

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-358215

(43)Date of publication of application : 26.12.2000

(51)Int.Cl.

H04N 5/91  
G11B 20/10  
G11B 27/00  
G11B 27/031

(21)Application number : 2000-128527

(71)Applicant : SAMSUNG ELECTRONICS CO LTD

(22)Date of filing : 27.04.2000

(72)Inventor : PARK SUNG-WOOK  
MOON SEONG-JIN  
OH YOUNG-NAM

(30)Priority

Priority number : 99 9915098  
99 9926050

Priority date : 27.04.1999  
30.06.1999

Priority country : KR

KR

(54) DATA STRING TEMPORARY ERASING METHOD, ITS RETRIEVAL METHOD, ITS RESTORING METHOD, ITS PERMANENT ERASING METHOD AND RECORDING MEDIUM FOR STORING ADDITIONAL INFORMATION FOR RESTORING DATA STRING SEPARATED BY TEMPORARY ERASURE

(57)Abstract:

PROBLEM TO BE SOLVED: To temporarily erase a data string having a packet structure through the use of locating information for restoring its original state by storing a second piece of additional information which indicates that the data string to which a first piece of additional information is added and the advanced data string are on continuous data string before erasure.

SOLUTION: One piece of cell information is separated into three pieces of cell information in order to temporarily erase a part of the data string and, then, a TE flag indicating that the temporary erasure is executed is set in an intermediate cell among them for restoration afterwards. The reason why the TE flag is set is to restore the temporarily erased part afterwards. Then the TE flag indicating the temporary erasure and a CT flag indicating that the data string and the advanced data string are the one continuous data string before the erasure are stored in data string information. The CT flag means that the data string related to the CT flag and the advance data string are one continuous data string before the temporary erasure. This means that synthesization with the advanced data string is possible.



## LEGAL STATUS

[Date of request for examination]

27.04.2000

[Date of sending the examiner's decision of

rejection]

[Kind of final disposal of application other than the  
examiner's decision of rejection or application  
converted registration]

[Date of final disposal for application]

[Patent number] 3541311

[Date of registration] 09.04.2004

[Number of appeal against examiner's decision of  
rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-358215

(P 2 0 0 0 - 3 5 8 2 1 5 A)

(43) 公開日 平成12年12月26日 (2000. 12. 26)

(51) Int. Cl. <sup>7</sup>	識別記号	F I	ターマコード <sup>*</sup> (参考)
H04N 5/91		H04N 5/91	N
G11B 20/10		G11B 20/10	G
27/00		27/00	A
27/031		27/02	B
			Z
審査請求 有 請求項の数22 O L (全23頁)			

(21) 出願番号	特願2000-128527 (P 2000-128527)	(71) 出願人	390019839 三星電子株式会社 大韓民国京畿道水原市八達区梅灘洞416
(22) 出願日	平成12年4月27日 (2000. 4. 27)	(72) 発明者	朴 成▲ウォク▼ 大韓民国ソウル特別市瑞草区瑞草3洞1595-2番地センチュリオフィステル2棟1207号
(31) 優先権主張番号	1 9 9 9 1 5 0 9 8	(72) 発明者	文 誠辰 大韓民国ソウル特別市永登浦区大林2洞1080-51番地
(32) 優先日	平成11年4月27日 (1999. 4. 27)	(74) 代理人	100064908 弁理士 志賀 正武 (外1名)
(33) 優先権主張国	韓国 (K R)		
(31) 優先権主張番号	1 9 9 9 2 6 0 5 0		
(32) 優先日	平成11年6月30日 (1999. 6. 30)		
(33) 優先権主張国	韓国 (K R)		

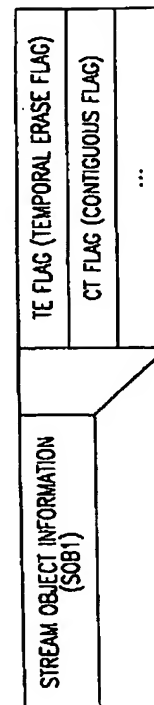
最終頁に続く

(54) 【発明の名称】 データ列の臨時削除方法、探索方法、復元方法、永久削除方法、及び臨時削除により分離されたデータ列を復元するための付加情報を貯蔵する記録媒体

(57) 【要約】

【課題】 データ列の臨時削除方法、探索方法、復元方法、永久削除方法及び臨時削除されたデータ列を復元するための付加情報を貯蔵する記録媒体を提供する。

【解決手段】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報の貯蔵されている記録媒体上に記録されたデータ列の一部を臨時削除する方法において、(a)臨時削除しようとする区間に対応して生成された複数のデータ列のための探索情報と連結情報のみを更新するが、臨時削除区間に該当するデータ列には臨時削除されたことを示す第1付加情報と、第1付加情報の付加えられているデータ列と先行するデータ列が臨時削除前に連続した1つのデータ列であったことを示す第2付加情報を貯蔵する段階を含むことを特徴とする。



## 【特許請求の範囲】

【請求項 1】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報の貯蔵されている記録媒体上に記録されたデータ列の一部を臨時削除する方法において、

(a) 臨時削除しようとする区間に対応して生成された複数のデータ列のための探索情報と連結情報のみを更新するが、臨時削除区間に該当するデータ列には臨時削除されたことを示す第 1 付加情報と、第 1 付加情報の付加えられているデータ列と先行するデータ列が臨時削除前に連続した 1 つのデータ列であったことを示す第 2 付加情報を貯蔵する段階を含む方法。

【請求項 2】 前記第 1 付加情報と第 2 付加情報は前記連結情報に含まれることを特徴とする請求項 1 に記載の方法。

【請求項 3】 前記第 1 付加情報と第 2 付加情報は前記探索情報に含まれることを特徴とする請求項 1 に記載の方法。

【請求項 4】 前記(a)段階は、

(a 1) 臨時削除により分離される各データ列の最初パケット到達時間と最後到達時間とを更新する段階と、

(a 2) 臨時削除により分離される各データ列の連結目録数を更新する段階と、

(a 3) 臨時削除により分離される各データ列が構成するデータ列ユニットを区分する段階と、

(a 4) 臨時削除により分離される各データ列の境界にあるデータ列ユニットの持続時間値を修正する段階と、

(a 5) 臨時削除区間に該当するデータ列のための第 1 付加情報と第 2 付加情報とをセットし、一部が臨時削除区間に含まれ前記臨時削除区間に該当するデータ列の後続データ列のための第 2 付加情報をセットする段階と、

(a 6) 臨時削除により分離される各データ列に対応するセルにおいて最初パケット到達時間と最後パケット到達時間とを更新する段階とを含む請求項 1 に記載の方法。

【請求項 5】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報の貯蔵されている記録媒体上に記録された複数のデータ列を探索する方法において、

(a) 所望の探索地点の到達時間を設定し、前記所望の探索地点に属するデータ列の最初パケット到達時間の整数部分を初期値として設定する段階と、

(b) 前記初期値にデータ列ユニットの持続時間を累算して累算値を提供する段階と、

(c) 設定された探索地点の到達時間が前記累算値より小さいか、同一になるまで前記(b)段階を繰返して行う段階とを含む方法。

【請求項 6】 パケットデータの到着した時間情報が付加えられた複数のデータ列を探索するための探索情報、

前記探索情報を実際のデータ列に連結させる連結情報が貯蔵されている記録媒体上に編集により割れたデータ列を元の状態に復元する方法において、

(a) 編集区間に対応して生成された複数のデータ列を元のデータ列に復元するために探索情報と連結情報のみを更新するが、該当データ列が編集されたことを示す第 1 付加情報と、編集されたデータ列の前に先行するデータ列とは編集前に連続する 1 つのデータ列であったことを示す第 2 付加情報とを無効化させる段階を含む方法。

【請求項 7】 前記編集は臨時削除であることを特徴とする請求項 6 に記載の方法。

【請求項 8】 前記(a)段階は、

(a 1) 統合される元のデータ列の最初パケット到達時間と最後到達時間とを、各々臨時削除区間の最初部分の属するデータ列の最初パケット到達時間と臨時削除区間の最後部分の属するデータ列の最後パケット到達時間に更新する段階と、

(a 2) 統合される元のデータ列の連結目録項目数は臨時削除により分離される各データ列の連結目録数を全て足した値に更新する段階と、

(a 3) 臨時削除により分離される各データ列を構成するデータ列ユニットを統合する段階と、

(a 4) 統合される元のデータ列の各データ列ユニットの持続時間値を修正する段階と、

(a 5) 統合される元のデータ列のための第 1 付加情報と第 2 付加情報とをリセットする段階と、

(a 6) 統合される元のデータ列に対応するセルにおいて最初パケット到達時間と最後パケット到達時間とを、各々臨時削除区間の最初部分の属するセルの最初パケット到達時間と臨時削除区間の最後部分の属するセルの最後パケット到達時間に更新する段階とを含む請求項 7 に記載の方法。

【請求項 9】 パケットデータが到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列と連結させる連結情報の貯蔵されている記録媒体上に臨時削除により分離された各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法において、

(a) 臨時削除により分離された各データ列の境界部分で先行するデータ列の最後データ列ユニットと関連した持続時間を更新する段階と、

(b) 臨時削除により分離された各データ列の境界部分で後続データ列の最初データ列ユニットと関連した持続時間を更新する段階とを含む方法。

【請求項 1 0】 前記(a)段階は、

(a 1) 臨時削除区間に該当するデータ列の最初データ列ユニットの最初パケット到達時間を丸めて整数からなる第 1 値と、前記先行するデータ列の最後データ列ユニットの最後パケット到達時間の整数部分のみ取って第 2 値を提供する段階と、

10

20

30

40

50

(a 2) 前記第 1 値から第 2 値を減算して第 1 減算結果を提供し、前記先行データ列の最後データ列ユニットの持続時間から前記第 1 減算結果を減算して第 2 減算結果を提供する段階と、

(a 3) 前記第 2 減算結果から単位値だけ増加させて修正された先行データ列の最後データ列ユニットと関連した持続時間を提供する段階とを含む請求項 9 に記載の方法。

【請求項 1 1】 前記(b)段階は、

(b 1) 前記後続データ列の最初データ列ユニットの最初

パケット到達時間が整数か否かを判断する段階と、  
(b 2) 前記後続データ列の最初データ列ユニットの最初パケット到達時間が整数でなければ、前記後続データ列の最初データ列ユニットの持続時間から単位値だけ増加させて修正し、そうでなければそのまま後続データ列の最初データ列ユニットの持続時間を提供する段階とを含む請求項 9 に記載の方法。

【請求項 1 2】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際データ列と連結させる連結情報の貯蔵されている記録媒体上に編集により割れたデータ列を復元する時、各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法において、

(a) データ列を統合する時、統合される各データ列の境界部分で先行データ列の最後データ列ユニットと関連した持続時間を更新する段階と、

(b) データ列を統合する時、統合される各データ列の境界部分で後続データ列の最初データ列ユニットと関連した持続時間を更新する段階とを含む方法。

【請求項 1 3】 前記(a)段階は、

(a 1) 先行データ列の最後データ列ユニットの最後パケット到達時間の整数部分に単位値を足した第 1 値と、後続データ列の最初データ列ユニットのパケット到達時間から整数部分のみを取った第 2 値を提供する段階と、

(a 2) 前記第 2 値から先行データ列の最後データ列ユニットの持続時間の差を求め、前記先行データ列の最後データ列ユニットの持続時間に前記差を加算して加算結果を提供する段階と、

(a 3) 後続データ列の最初データ列ユニットのパケット到達時間が整数でなければ前記加算結果から単位値だけ増加させて修正した先行データ列の最後データ列ユニットの持続時間を提供し、そうでなければそのまま先行データ列の最後データ列ユニットの持続時間を提供する段階とを含む請求項 1 2 に記載の方法。

【請求項 1 4】 前記(b)段階は、

(b 1) 前記後続データ列の最初データ列ユニットの最初パケット到達時間が整数か否かを判断する段階と、

(b 2) 前記後続データ列の最初データ列ユニットの最初パケット到達時間が整数でなければ、前記後続データ列

の最初データ列ユニットの持続時間から単位値だけ減少させ修正し、そうでなければそのまま後続データ列の最初データ列ユニットの持続時間を提供する段階とを含む請求項 1 2 に記載の方法。

【請求項 1 5】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列と連結させる連結情報の貯蔵されている記録媒体上に一部のデータ列を永久削除する方法において、

(a) 永久削除しようとする区間に該当するデータ列のための探索情報を更新する段階と、

(b) 前記連結情報を更新するが、永久削除されない正常データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する段階とを含む方法。

【請求項 1 6】 前記(a)段階では永久削除しようとする区間に対応する探索情報を削除することを特徴とする請求項 1 5 に記載の方法。

【請求項 1 7】 前記(b)段階は、

(b 1) 前記永久削除可能なデータ列ユニットと関連した連結情報を削除する段階と、

(b 2) 前記永久削除区間に該当するデータ列と先行データ列の境界部分で前記先行データ列の最後データ列ユニットと関連した持続時間を更新する段階と、

(b 3) 前記永久削除区間に該当するデータ列と後続データ列の境界部分で前記後続データ列の最初データ列ユニットと関連した持続時間を更新する段階とを含む請求項 1 5 に記載の方法。

【請求項 1 8】 前記(b 2)段階は、

(b 2 1) 前記永久削除区間に該当するデータ列の最初データ列ユニットの最初パケット到達時間を丸めて整数からなる第 1 値と、前記先行データ列の最後データ列ユニットの最後パケット到達時間の整数部分のみ取って第 2 値を提供する段階と、

(b 2 2) 前記第 1 値から第 2 値を減算して第 1 減算結果を提供し、前記先行データ列の最後データ列ユニットの持続時間から前記第 1 減算結果を減算して第 2 減算結果を提供する段階と、

(b 2 3) 前記第 2 減算結果から単位値だけ増加させて修正された先行データ列の最後データ列ユニットと関連した持続時間を提供する段階とを含む請求項 1 7 に記載の方法。

【請求項 1 9】 前記(b 3)段階は、

(b 3 1) 前記後続データ列の最初データ列ユニットの最初パケット到達時間が整数か否かを判断する段階と、

(b 3 2) 前記後続データ列の最初データ列ユニットの最初パケット到達時間が整数でなければ、前記後続データ列の最初データ列ユニットの持続時間から単位値だけ増加させて修正し、そうでなければそのまま後続データ列の最初データ列ユニットの持続時間を提供する段階とを含む請求項 1 7 に記載の方法。

【請求項 2 0】 パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報と前記探索情報を実際のデータ列と連結させる連結情報が貯蔵され、  
臨時削除区間に該当するデータ列には、臨時削除されたことを示す第 1 付加情報と、第 1 付加情報の付加えられているデータ列と先行データ列とが臨時削除以前に連続して 1 つのデータ列であったことを示す第 2 付加情報の貯蔵される記録媒体。

【請求項 2 1】 前記臨時削除しようとする区間に対応して生成された複数のデータ列のための探索情報と連結情報とが更新され、前記第 1 付加情報と第 2 付加情報は前記連結情報に含まれることを特徴とする請求項 2 0 に記載の記録媒体。

【請求項 2 2】 前記臨時削除しようとする区間に対応して生成された複数のデータ列のための探索情報と連結情報とが更新され、前記第 1 付加情報と第 2 付加情報は前記探索情報に含まれることを特徴とする請求項 2 0 に記載の記録媒体。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】本発明はパケット構造を有するデータの編集及び/または再生分野に係り、特にデータ列の臨時削除方法、探索方法、復元方法、永久削除方法、及び臨時削除により分離されたデータ列を元のデータ列に復元するための付加情報を貯蔵する記録媒体に関する。

【 0 0 0 2 】

【従来の技術】一般に一つのコンテンツは 1 つのデータ列(Stream Object:SOB)で形成され、1 回のデータ列は複数の一定の単位に分けられたデータ列ユニット(Stream Object Unit:SOBU)で管理される。この時のデータ列は、例えば、使用者が 1 つの記録を開始して終了するまでに記録されたデータを意味し、具体的にはドラマー遍または映画一遍が 1 つのデータ列の形態として記録される。従って、コンテンツの代わりにプログラムという表現が使用できるので、以下プログラムとして説明する。

【 0 0 0 3 】図 1 は既に記録されたデータ列に対して所望の部分のデータを読み出すのに用いられる探索情報とデータとの間の関係を示す例である。何れのプログラムを使用者が録画する場合、使用者の目には 1 つのプログラムのみが存在すると見られるが内部的には使用者に意味のある探索手段を提供するセル(Cell)、セルにある情報を実際にデータと連結させるデータ列情報(SOB information:SOBI)が存在する。

【 0 0 0 4 】特に、データ列は貯蔵媒体の物理的な性質、再生/編集装置の性能及び/または管理の便宜と効率を高めるためにデータを一定の単位(データ列ユニット:SOBU)に束ねて管理するが、データ列ユニットへの接近

時用いられる情報がデータ列情報の連結目録(MAPping List:MAPL)に貯蔵される。SOBUの番号は 1 から始まり、連結目録は複数個の項目を有しうる。この項目はデータ列ユニット(SOBU)の持続時間を示すIAPAT(Incremental AP Arrival Time)という情報を有する。

【 0 0 0 5 】貯蔵装置に記録されたプログラムを任意に接近可能にするために、一般に接近しようとするプログラムに対する探索情報を別に作って使用する。探索情報としては、プログラム内でデータの位置情報、プログラム再生時間とプログラム記録時間が通用され、記録されているプログラムが映画なら映画の特徴的な場面等も使われるが、ここではプログラム記録時間を探索情報の一例として使用する。

【 0 0 0 6 】探索情報は応用分野によって多様な形式を有するが、通常の場合には情報量を減らし、迅速な探索を可能にするためにプログラムを構成するデータを 1 つずつ指示しなく、大きく束ねて指示する。このような束は通常階層的に構成されることもあり、図 1 では 3 つの階層で構成した例を示す。階層的構造で探索情報が構成される場合、上位階層の情報は直下の正常な下位階層の情報からなる。プログラムを構成する探索情報の上位階層としてはセルが存在し、その下にはデータ列、データ列ユニット順に存在する。探索情報の各階層は相互多様な関係を有しうる、ここでは次の特定の場合を仮定する。

【 0 0 0 7 】プログラムは 1 つ以上のセルよりなる。セルは 1 つのデータ列で構成される。データ列は 1 つ以上のデータ列ユニットで構成される。データ列ユニットは 1 つ以上のデータで構成される。図 1 ではProgram\_orgがCell\_orgで、Cell\_orgはSOBで、SOBはSOBU(1)、SOBU(2)、SOBU(3)で、各SOBUはデータで、各々構成された場合を示す。即ち、Program\_orgは自分のセルを記憶する情報構造を有しており、セルは自分のSOBを記憶する情報構造を、SOBはSOBUを記憶する情報構造の連結目録(MAPL)を有している。

【 0 0 0 8 】また、探索情報が管理するデータ列の範囲を示すために、セルの最初データ到達時間SC\_S\_APATとセルの最後データ到達時間SC\_E\_APAT及びデータ列の最初データ到達時間SOB\_S\_APATとデータ列の最後データ到達時間SOB\_E\_APATとを各々セル情報とデータ列情報とに含める。臨時削除(Temporal Erase:TE)フラグはデータ列が臨時削除されたことを表示する。ここで、臨時削除とは、使用者がプログラム全体或は一部を削除した後、これを復元できる削除を意味する。逆に復元できない削除を永久削除という。

【 0 0 0 9 】MAPLの各項目は対応されるSOBUの持続時間を示すIAPAT(Incremental AP Arrival Time)で示す。ここで、APはアプリケーションパケット(Application Packet)であって、パケット構造を有するデータを意味する。SOBUの持続時間のIAPATは下記のように定義され

る。

【0010】時間で表現された探索情報は応用分野によってそれなりの精度を有している。MPEG(Moving Picture Expert Group)-2 システムの場合は 27MHz 単位で時間をカウントして使用する。図 1 では 0.1 単位でデータが表示され、SOB は探索情報の MAPL の大きさを縮めるために 1.0 単位に時間を示すと仮定したことである。これを 48 ビット(6 バイト)大きさのレジスタを用いて時間を示す場合に適用すれば、図 2 に示されたようにポイント b18 以上が少数点以上に該当し、b18 未満が少数点以下に該当する。この少数点の直上の b18 に該当する位置を MTU\_SHFT と表示する。

【0011】また、図 1 において MAPL の各項目は 3、1、1 という値を有するが、これは MAPL に対する次の定義に立脚したものである。参考に、毎 SOBU の IAPAT を決定する方法は SOB において SOBU の位置によって、即ち、SOBU が最後 SOBU 否かに応じて異なる方法が使われる。例えば、1 つの SOB に M 個の SOBU がある場合、最後 SOBU を除いた i 番目の SOBU(i) に対し、IAPAT を最初から i 番目まで累算したのが SOBU(i+1) の最初 AP 到達時間より 1 以上大きくてはいけない。この際、IAPAT は 1.0 単位に表現される整数と仮定し、累算の初期値は 0 とする。これを数式で表せば次の通りである。

【0012】[数 1]

$$\text{SOBU\_S\_APAT}(i+1) \leq \text{SUM\_IAPAT}(i) < \text{SOBU\_S\_APAT}(i+1) + 1$$

ここで、SUM\_IAPAT(i) は該当 SOBU、即ち、SOBU(i) を含んで先行する全ての SOBU の IAPAT 値を累算したことを示し、SOBU\_S\_APAT(i+1) は SOBU(i+1) の最初 AP の到達時間を示す。M 個の SOBU がある場合、最後 SOBU の SOBU(M) に対して IAPAT を最初から M 番目まで累算したのが SOBU(M) の最後 AP 到達時間より大きい、SOBU(M) の最後 AP 到達時間より 1 を超過してはいけない。この際、IAPAT は 1.0 単位で表現される整数と仮定し、累算の開始は“0”にする。

【0013】[数 2]

$$\text{SOBU\_E\_APAT}(M) < \text{SUM\_IAPAT}(M) \leq \text{SOBU\_S\_APAT}(M) + 1$$

ここで、SUM\_IAPAT(M) は該当 SOBU、即ち、SOBU(M) を含んで先行する全ての IAPAT を累算した値を示し、SOBU\_E\_APAT(M) は SOBU(M) の最後 AP の到達時間を示す。前記数式 1 と数式 2 の概念を図 3 に基づいて詳しく説明する。例えば、IAPAT(1)、IAPAT(2) と IAPAT(6) を求めると次のようである。

【0014】図 3 において、SOBU(1) の場合、SOBU(1) の IAPAT を累算した結果、即ち SOBU(1) の値が SOBU(2) の最初 AP 到達時間より大きい、SOBU(2) の最初 AP 到達時間より 1 以上大きくてはいけない。即ち、累算結果が 1.9 より大きい、SOBU(1) の IAPAT は 2 となる。

2 となる。

【0015】SOBU(2) の場合、SOBU(1) の IAPAT と SOBU(2) の IAPAT とを累算した結果が SOBU(3) の最初 AP 到達時間より大きい、SOBU(3) の最初 AP 到達時間より 1 以上大きくてはいけない。即ち、累算結果が 5.5 より大きい、SOBU(2) の IAPAT は 6 となる。

【0016】このような方法で最初と中間 SOBU の IAPAT が分かり、図 4A に示された場合は、境界条件なので特に注意すべきである。図 4A に示された SOBU(2) の IAPAT を計算すれば、SOBU(3) の最初 AP 到達時間が整数 5.0 の場合 SOBU(2) の IAPAT まで累算した結果は 6 でなく 5 である。一方、図 3 に示された SOBU(6) の場合は SOB の最初 AP 到達時間の整数部分と SOBU(6) までの IAPAT を累算した結果が SOBU(6) の最後 AP 到達時間より大きい、SOBU(M) の最後 AP 到達時間より 1 を超過してはいけない。即ち、累算結果が 10.8 より大きい、SOBU(5) まで IAPAT を累算した結果が 10 になるので SOBU(6) の IAPAT は 1 となる。

【0017】このような方法で 1 つの SOB から最後 SOBU の IAPAT がわかり、図 4B に示された場合は境界条件なので特に注意すべきである。図 4B に示された SOBU(6) の IAPAT を計算すれば、SOBU(6) の最後 AP 到達時間が整数 11.0 の場合 SOBU(6) の IAPAT まで累算した結果は 11 でなく 12 である。一方、一般にプログラムが臨時削除される場合 TE フラグで表示すると約束されている。従って、プログラムの一部を臨時削除する場合、データ列を分離して臨時削除されたデータ列と削除されていないデータ列とに区分し、臨時削除されたデータ列には TE フラグをセットする。このような作業を行うのはデータを実際に消さなく使用者とデータとを連結する探索情報のみを変更することによって、後で削除された部分を復元可能にするためである。

【0018】しかし、このようにデータ列を分離するにおいて、探索情報を生成する方法が提示されていない。また、臨時削除された部分を表示するのが TE フラグだけなので後で復元するに当たって TE フラグをリセットする方法を採択すべきである。これら、セル間の読出は中間にリセット過程が挿入され、不連続的に行われることがある。

【0019】このように TE フラグをリセットして復元する方法を使用すれば、復元されたプログラムは 2 つ以上のセルを有することになる。これは臨時削除後、臨時削除された部分を復元してもセル間の動作に対して定義されていないために、図 1 に示されたように元のプログラムへの復帰ではないことを意味する。従って、使用者としては臨時削除後の復元は元の状態への復帰を意味する

ので臨時削除によって分離されたデータ列を完全に復元する方法が必要となる。

#### 【 0 0 2 0 】

【発明が解決しようとする課題】従って、本発明の目的は、パケット構造を有するデータ列において元の状態に復元するための探索情報を用いて臨時削除する方法を提供することにある。本発明の他の目的は、複数のデータ列に分離されている時探索情報を用いて高速探索する方法を提供することにある。本発明のさらに他の目的は、臨時削除時複数個に分離されたデータ列を完全復元する

方法を提供することにある。

【 0 0 2 1 】本発明のさらに他の目的は、臨時削除により分離された各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法を提供することにある。本発明のさらに他の目的は、編集により割れたデータ列を元の状態に復元する時、各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法を提供することにある。

【 0 0 2 2 】本発明のさらに他の目的は、所望の部分のデータ列を永久削除する方法を提供することにある。本発明のさらに他の目的は、臨時削除時、多数個に分離されたデータ列を完全復元するための付加情報を貯蔵する記録媒体を提供することにある。

#### 【 0 0 2 3 】

【課題を解決するための手段】前記目的を達成するために本発明に係る臨時削除方法は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報の貯蔵されている記録媒体上に記録されたデータ列の一部を臨時削除する方法において、臨時削除しようとする区間に対応して生成された複数のデータ列のための探索情報と連結情報のみを更新するが、臨時削除区間に該当するデータ列には臨時削除されたことを示す第 1 付加情報と、第 1 付加情報の付加えられているデータ列と先行するデータ列が臨時削除前に連続した 1 つのデータ列であったことを示す第 2 付加情報を貯蔵する段階を含むことを特徴とする。

【 0 0 2 4 】本発明に係る探索方法は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報の貯蔵されている記録媒体上に記録された複数個のデータ列を探索する方法において、所望の探索地点の到達時間を設定し、前記所望の探索地点に属するデータ列の最初パケット到達時間の整数部分を初期値として設定する段階と、前記初期値にデータ列ユニットの持続時間を累算して累算値を提供する段階と、設定された探索地点の到達時間が前記累算値より小さいか、同一になるまで前記 (b) 段階を繰返して行う段階とを含むことを特徴とする。

【 0 0 2 5 】本発明に係る復元方法は、パケットデータ

の到着した時間情報が付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列に連結させる連結情報が貯蔵されている記録媒体上に編集により割れたデータ列を復元する方法において、編集区間に対応して生成された複数のデータ列を元のデータ列に復元するために探索情報と連結情報のみを更新するが、該当データ列が編集されたことを示す第 1 付加情報と、編集されたデータ列の前に先行するデータ列とは編集前に連続する 1 つのデータ列であったことを示す第 2 付加情報とを無効化させる段階を含むことを特徴とする。

【 0 0 2 6 】本発明に係る臨時削除時連結目録更新方法は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列と連結させる連結情報の貯蔵されている記録媒体上に臨時削除により分離された各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法において、臨時削除により分離された各データ列の境界部分で先行するデータ列の最後データ列ユニットと関連した持続時間を更新する段階と、臨時削除により分離された各データ列の境界部分で後続データ列の最初データ列ユニットと関連した持続時間を更新する段階とを含むことを特徴とする。

【 0 0 2 7 】本発明に係る復元時連結目録更新方法は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列と連結させる連結情報の貯蔵されている記録媒体上に編集により割れたデータ列を復元する時、各データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する方法において、データ列を統合する時、統合される各データ列の境界部分で先行データ列の最後データ列ユニットと関連した持続時間を更新する段階と、データ列を統合する時、統合される各データ列の境界部分で後続データ列の最初データ列ユニットと関連した持続時間を更新する段階とを含むことを特徴とする。

【 0 0 2 8 】本発明に係る永久削除方法は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報、前記探索情報を実際のデータ列と連結させる連結情報の貯蔵されている記録媒体上に一部のデータ列を永久削除する方法において、永久削除しようとする区間に該当するデータ列のための探索情報を更新する段階と、前記連結情報を更新するが、永久削除されない正常データ列の境界部分に対応するデータ列ユニットのための各連結目録情報を更新する段階とを含むことを特徴とする。

【 0 0 2 9 】また、本発明に係る記録媒体は、パケットデータの到着した時間情報の付加えられた複数のデータ列を探索するための探索情報と前記探索情報を実際のデータ列と連結させる連結情報の貯蔵され、臨時削除区間



に該当するデータ列には臨時削除されたことを示す第 1 付加情報と第 1 付加情報の付加えられているデータ列と、先行データ列とが臨時削除以前に連続して 1 つのデータ列であったことを示す第 2 付加情報の貯蔵されることを特徴とする。

#### 【 0 0 3 0 】

【発明の実施の形態】以下、添付した図面に基づき本発明に係るデータ列の臨時削除方法、探索方法、復元方法、永久削除方法及び臨時削除により分離されたデータ列を復元するための付加情報を貯蔵する記録媒体の望ましい実施例を説明する。図 5 はデータ列の一部を臨時削除する場合、削除範囲を決定する過程を示す一例である。使用者としては貯蔵されたプログラムを任意に、できるだけ最小の単位まで調整しようとする。即ち、使用者が一部を臨時削除しようとするプログラムが映画の場合、一枚単位の編集を希望する。この場合、使用者が調整する精度が、実際に編集装置がデータを処理する単位のデータ列ユニット(SOBU)より小さいので削除範囲を決定する過程で格別な注意が必要である。

【 0 0 3 1 】図 5 は使用者がデータ列ユニット(SOBU)より小さな単位の 0. 1 の精度で 1. 5 から 3. 8 まで部分削除を行う場合である。探索情報を用いて使用者が臨時削除しようと指定した時間を探し、これを基にして 1 つのセルを 3 つのセルに分離する過程を示す。使用者がプログラムの中間に該当する  $x-y$  に互った時間区間を臨時削除することを示しており、プログラムの最初部分削除及び/または最後部分削除が全て含まれることができる。

【 0 0 3 2 】  $x$  から  $y$  を削除するために、先に  $x$  の属する SOBU と  $y$  の属する SOBU を探す。図 5 では  $x$  は SOBU (1) に、  $y$  は SOBU (3) に属することが分かる。従って、 1 つのセルを 3 つのセルに分離するに当って、最初セルを SOBU (1) 以下の SOBU を含む SOB で、 3 番目のセルを SOBU (3) 以上のセルを含む SOB で、 2 番目のセルを前述しなかった SOBU を含む SOB で構成する。その理由は探索情報を構成する規則において、セルは 1 つのデータ列で、 1 つのデータ列は多数個の SOBU で構成されるように規定するからである。即ち、最初セルは SOBU (1) を有する SOB 1 で、 3 番目のセルは SOBU (3) を有する SOB 3 で、 2 番目のセルは SOBU (2) を有する SOB 2 で構成される。

【 0 0 3 3 】図 6 は臨時削除で分離されたデータ列の一例であって、削除しようとする部分は斜線で表した  $x$  から  $y$  までで、セル 2 で表されている。データ列の一部を臨時削除するために 1 つのセル情報を 3 つのセル情報に分離した後、このうち中間のセルに今後復元のために臨時削除されたことを示す TE フラグをセットする。TE フラグをセットする理由はこのような作業(臨時削除)が行われた部分を今後復元可能にするためである。この際、最初セルは元のセルの最初から削除前までを、最後セルは削除された以後から元のセルの最後までを含む。セルの

探索情報とデータとを連結するデータ列情報は各々複数個のデータ列ユニットを有しうる。ここで、セル 1 は  $0.8-x(=1.5)$  までを示し、SOB 1 は  $0.8-1.9$  までを、セル 2 は  $2.1-3.3$  までを、SOB 2 は  $2.1-3.3$  までを、セル 3 は  $3.8(=y)-4.8$  までを、SOB 3 は  $3.5-4.8$  までを各々示す。

【 0 0 3 4 】セルの範囲とデータ列情報の範囲が一致しない理由は、SOB は SOBU 単位で時間を記録し、セルは SOB 範囲内で使用者が削除した時間を示すからである。かかる方法を通して SOB は SOBU 単位で SOB が管理でき、セルは使用者に使用者の操作内容が正しく反映されたことを示しうる。

【 0 0 3 5 】従って、図 6 は  $x$  から  $y$  に臨時削除過程を通して 3 つのセルに分離されたプログラムの探索情報を示す。セル 2 は臨時削除される部分を束ねてセルという情報構造を表し、臨時削除されたことを表示するために TE フラグをセットする。このような探索情報を有している場合、再生装置はセル 1 とセル 3 とを再生してこれらを連続的に再生する場合セル 1 とセル 3 との境界で一般にリセット過程を行う。セル 2 は臨時削除された状態なので再生されない。 $x$  から  $y$  までの時間を臨時削除する過程で 1 つのセルが 3 つのセルに分離される。この際、分離された各セルのそれぞれの探索情報は図 7 のテーブルに示されている。

【 0 0 3 6 】図 7 に示されたテーブルにおいて、AP はパケットという構造を有する応用で使用するデータである。IAPAT (Incremental AP arrival time) は AP 到達時間を累算して解かるようにした形態の AP 到達時間を、IAPAT (i) は SOBU (i) の IAPAT、SOB\_S\_APAT (SOB の最初 AP 到達時間) はデータ列において最初位置にある開始データの到達時間を、SOB\_E\_APAT (SOB の最後 AP 到達時間) はデータ列において最後位置にある最後データの到達時間、SOB\_S\_SOBU (SOB において最初 SOBU) はデータ列において最初位置にある開始データ列ユニットの番号を示す。MAPL\_ENT\_Ns (MAPL 項目数) はデータ列に属した IAPAT 数であってデータ列ユニットの数と同一で、MAPL [IAPATs] (MAPL 項目) は IAPAT の集合で、SOB\_TY (SOB type) はデータ列のタイプであって TE フラグと CT (Contiguous) フラグなどが位置し、SC\_S\_APAT (Cell の最初 AP 到達時間) はセルにおいて最初位置にある開始データの到達時間で、SC\_E\_APAT (セルの最後 AP 到達時間) はセルにおいて最後位置にある最後データの到達時間を意味する。ここで、CT フラグについては図 1 5 において詳しく説明する。

【 0 0 3 7 】臨時削除時探索情報を生成する過程は次の通りである。まず、SOBU 番号は 1 から始まる。 $ix$  は  $x$  の属する SOBU の番号を意味する。即ち、SOBU (ix) は  $x$  の属する SOBU を意味する。但し、 $x$  が SOBU の最初 AP 到達時間と同一な場合、 $ix$  は  $x$  の属する SOBU の前 SOBU 番号となる。 $iy$  は  $y$  の属する SOBU 番号を意味する。即ち、SOBU (iy) の属する SOBU を意味する。但し、 $y$  が SOBU の最後 AP 到

達時間と同一な場合、 $iy$ は $y$ の属するSOBUの後SOBU番号となる。従って、図6においてSOBU( $ix$ )はSOBU(1)で、SOBU( $iy$ )はSOBU(3)である。参考にSOBU( $ix+1$ )はSOBU(2)で、SOBU( $iy-1$ )はSOBU(2)である。

【0038】第1段階:SOB1の最初AP到達時間はSOB\_orgの最初AP到達時間をコピーして使用する。

第2段階:SOB2の最初AP到達時間はSOBU( $ix+1$ )の最初AP到達時間とする。

第3段階:SOB3の最初AP到達時間はSOBU( $iy$ )の最初AP到達時間とする。

第4段階:SOB1の最後AP到達時間はSOBU( $ix$ )の最後AP到達時間を使用する。

第5段階:SOB2の最後AP到達時間はSOBU( $iy-1$ )の最後AP到達時間を使用する。

第6段階:SOB3の最後AP到達時間はSOB\_orgの最後AP到達時間をコピーして使用する。図8は第1段階乃至第6段階の過程を示す例である。

第7段階:SOB1の最初SOBUはSOB\_orgの最初SOBUとして設定される。

第8段階:SOB2の最初SOBUはSOBU( $ix+1$ )として設定される。

第9段階:SOB3の最初SOBUはSOBU( $iy$ )で(に)設定される。

【0039】第7段階乃至第9段階を図6に適用すれば、SOB1、SOB2、SOB3が全て1つのSOBUを有するのでこれらが直ちにSOBの最初SOBUとなる。従って、図7のテーブルに示されたようにSOB1、SOB2、SOB3の最初SOBUは各々SOBU(1)、SOBU(2)、SOBU(3)となる。

第10段階:SOB1のMAPL項目数は $ix$ となる。

第11段階:SOB2のMAPL項目数は[SOBU( $iy-1$ )番号-SOB U( $ix+1$ )番号+1]と決められる。

第12段階:SOB3のMAPL項目数は[SOBのMAPL項目数-SOB U( $iy$ )番号+1]と決められる。

【0040】第10段階乃至第12段階を図6に適用すればSOB1、SOB2、SOB3のMAPL項目数は各々1個となる。これは、臨時削除後3つに分離された後、SOB1、SOB2、SOB3は全てSOBUを1つずつ有していることを意味する。

第13段階:SOB1、SOB2、SOB3のIAPATを決定する。これについては、図10A、図10B、図11A、図11B、図12及び図13を結び付けて説明する。1つのデータ列(SOB)が分離される場合、分離される境界にあるSOBUのIAPAT値のみが変化する。その理由はIAPATを決定する規則がSOBにおけるSOBU位置に応じて別に適用されるからである。

【0041】即ち、図9では、分離前SOB\_orgがSOB1、SOB2、SOB3に分離された後IAPATが変化した形態を示している。SOBU(1)がSOB1の最後SOBUとなり、SOBU(2)がSOB2の最初SOBUとなる。また、SOBU(2)はSOB2の最後SOBUとなり、SOBU(3)はSOB3の最初SOBUとな

る。従って、SOBU(1)のIAPATは1.0が減った2.0となり、SOBU(2)のIAPATは1.0が増えた2.0となり、SOBU(3)のIAPATは1.0が増えた2.0となる。

【0042】これに対して、図10A、図10B、図11A及び図11Bに示した流れ図を結び付けてさらに詳しく説明する。SOBU( $ix$ )の最後AP到達時間、SOBU( $ix+1$ )の最初AP到達時間、SOBU( $iy-1$ )の最後AP到達時間、SOBU( $iy$ )の最初AP到達時間は既知のことと仮定する。臨時削除によるデータ列分離後SOB1とSOB2の境界において、SOB1の最後SOBUのSOBU( $ix=1$ )と関連したIAPATとSOB2の最初SOBUのSOBU( $ix+1=2$ )と関連したIAPATはデータ列の内部における相対的な位置が変更される。従って、これらの値を修正すべきである。

【0043】まず、SOB1の最後SOBUのSOBU( $ix$ )のIAPATを修正する方法の流れ図は図10Aに示されたようである。まず、SOB2の最初SOBUのSOBU( $ix+1=2$ )の最初AP到達時間を丸めて整数にし、これをpreEnd\_high(=3)という変数に貯蔵し、SOB1の最後SOBUのSOBU( $ix=1$ )最後AP到達時間の整数分のみ取ったことをpreEnd\_APAT\_high(=1)という変数に貯蔵し、SOB1の最後SOBUのSOBU( $ix$ )のIAPATをpreEnd\_IAPAT(=3)という変数に貯蔵する(S101段階)。preEnd\_high値とpreEnd\_APAT\_high値との差(delta=2)を求め、preEnd\_IAPAT(=3)から差(delta=2)を引き、その減算結果(=1)から1だけ増加させ、preEnd\_IAPATを修正する(S102段階)。修正されたpreEnd\_IAPAT(=2)をSOBU( $ix$ )のIAPATに貯蔵する(S103段階)。従って、SOBU( $ix=1$ )のIAPATは2となる。

【0044】次いで、SOB2の最初SOBUのSOBU( $ix+1$ )のIAPATを修正する方法は、図11Aに示されたようである。まず、SOB2の最初SOBUのSOBU( $ix+1=2$ )の最初AP到達時間をsucStart\_APAT(=2.1)という変数に貯蔵し、SOB2の最初SOBUのSOBU( $ix+1$ )のIAPATをsucStart\_IAPAT(=1)という変数に貯蔵する(S121段階)。sucStart\_APATが整数か否かを判断し(S122段階)、もしsucStart\_APATが整数でなければsucStart\_IAPATを1だけ増加させ修正する(S123段階)。修正されたsucStart\_IAPAT(=2)をSOBU( $ix+1$ )のIAPATに貯蔵したり、S122段階においてsucStart\_APATが整数ならばS121段階で設定されたsucStart\_IAPAT(=1)をそのままSOBU( $ix+1$ )のIAPATに貯蔵する(S124段階)。従って、SOBU( $ix+1=2$ )のIAPATは2となる。

【0045】図12はSOB1とSOB2にデータ列が分離される場合、図10Aと図11Aに示された流れ図で使用した変数が指示する値を図式化している。同様に、臨時削除によりデータ列が分離される時SOB2とSOB3の境界でも、SOB2の最後SOBUのSOBU( $iy-1=2$ )と関連したIAPATとSOB3の最初SOBUのSOBU( $iy=3$ )と関連したIAPATはデータ列の内部における相対的な位置が変更される。これらの値を図10B及び図11Bに示されたように修正すべきである。

【 0 0 4 6 】 まず、SOB 2 の最後SOBUのSOBU(iy-1)のIAPATを修正する方法の流れ図は図 1 0 bに示されたようである。先にSOB 3 の最初SOBUのSOBU(iy=3)の最初AP到達時間を丸めて整数にし、これをpreEnd\_high(=4)という変数に貯蔵し、SOB 2 の最後SOBUのSOBU(iy-1=2)の最後AP到達時間の整数分のみ取ったものをpreEnd\_IAPAT\_high(=3)という変数に貯蔵し、SOB 2 の最後SOBUのSOBU(iy-1)のIAPATをpreEnd\_IAPAT(=2)という変数に貯蔵する(S 1 1 1 段階)。preEnd\_high値とpreEnd\_IAPAT\_high値との差(delta=1)を求め、preEnd\_IAPAT(=2)から差(delta=1)を引き、その減算結果(=1)から1だけ増加させpreEnd\_IAPATを修正する(S 1 1 2 段階)。修正されたpreEnd\_IAPAT(=2)をSOBU(iy-1)のIAPATに貯蔵する(S 1 1 3 段階)。従って、SOBU(iy-1=2)のIAPATは2となる。

【 0 0 4 7 】 次いで、SOB 3 の最初SOBUのSOBU(iy)のIAPATを修正する方法の流れ図は図 1 1 bに示されたようである。まず、SOB 3 の最初SOBUのSOBU(iy=3)の最初AP到達時間をsucStart\_IAPAT(=3.5)という変数に貯蔵し、SOB 3 の最初SOBUのSOBU(iy)のIAPATをsucStart\_IAPAT(=1)という変数に貯蔵する(S 1 3 1 段階)。sucStart\_IAPATが整数なのかを判断し(S 1 3 2 段階)、もしsucStart\_IAPATが整数でなければ、sucStart\_IAPAT(=1)を単位値(ここでは1)だけ増加させ修正する(S 1 3 3 段階)。修正されたsucStart\_IAPAT(=2)をSOBU(iy)のIAPATに貯蔵したり、S 1 3 2 段階でsucStart\_IAPATが整数ならS 1 3 1 段階で設定したsucStart\_IAPAT(=1)をそのままSOBU(iy)のIAPATに貯蔵する(S 1 3 4 段階)。従って、SOBU(iy=3)のIAPATは2となる。

【 0 0 4 8 】 図 1 3 はデータ列がSOB 2 とSOB 3 とに分離される場合に、図 1 0 bと図 1 1 bにおいて使用した変数が指示する値を図式化している。

第 1 4 段階: 臨時削除対象のSOB 2 のTEフラグとCTフラグとをセットする。臨時削除対象の次のデータ列のSOB 3 のCTフラグをセットする。

第 1 5 段階: SOB 1 に該当するセルにおい最初APの到達時間をSOB\_orgに該当するセルの最初AP到達時間とする。

第 1 6 段階: SOB 1 に該当するセルにおいて最後APの到達時間をxとする。

第 1 7 段階: SOB 2 に該当するセルにおいて最初APの到達時間をSOBU(ix+1)の最初AP到達時間とする。

【 0 0 4 9 】 第 1 8 段階: SOB 2 に該当するセルにおいて最後APの到達時間をSOBU(iy-1)の最後AP到達時間とする。

第 1 9 段階: SOB 3 に該当するセルにおいて最初APの到達時間をyとする。

第 2 0 段階: SOB 3 に該当するセルにおいて最後APの到達時間をSOB\_orgに該当するセルの最後AP到達時間とする。

このようにデータ列が分離され、修正された探索情報を用いてデータ列を探索する過程は次の通りである。SOB 3 を例として一般の場合を含む表現を使用すれば次の通りである。

【 0 0 5 0 】 例えば、図 1 3 に示されたように 4.5 にあるデータ、即ち、APを探そうとするなら、

第 1 段階: 変数ターゲットに 4.5 を貯蔵する。

第 2 段階: SOBの最初AP到達時間(SOB\_S\_IAPAT)の整数部分を累算の初期値とする。

第 3 段階: 変数iに 1 を貯蔵する。

第 4 段階: IAPAT(i)を累算する(sum=sum+IAPAT(i))。

第 5 段階: もし変数ターゲットが累算した値(sum)より小さいか、同じなら、変数ターゲットはSOBU(i)に属するものである。もし、変数ターゲットが累算した値(sum)より大きいと、iを1増加させて第4段階を行う。

【 0 0 5 1 】 ここで、探そうとするデータの属するデータ列ユニットを探し出す過程でSOBUの累積時間(IAPAT)を累算する時、初期値を“0”にしなく、データ列の最初データ到達時間の整数部として設定することによってIAPATが小さな値を保てる。もし、従来の方法の通り初期値を“0”にすれば、SOB 3 のIAPATは1でなく5という大きな値を有するべきである。このための回路具現時メモリがさらに多くのビット数を使用すべきことを意味する。

【 0 0 5 2 】 図 1 4 は単純復元で臨時削除により分離されたデータ列を復元する例であって、従来の探索情報を用いて臨時削除された部分を復元する例である。単純復元はTEフラグ設定で使用者に見られなかったデータ列においてTEフラグを無効化させる。即ち、セルの臨時削除を知らせるためにデータ列情報にあるTEフラグをリセットすることである。かかる方法を用いると、使用者はセル1、セル2、セル3を全て読出せる。しかし、具体的にセル1は0.8からx(=1.5)まで、セル2は2.1から3.3まで、セル3はy(=3.8)から4.8までを示すので、この範囲に属するデータのみが読出可能なので斜線部分は読出できない。また、元は1つのセルからなったプログラムが部分削除されて3つのセルに分離されたので、これを復元してもセルとセルとの間の動作に対する定義が必要である。これらセル間の読出は中間にリセット過程が挿入されて不連続的に行われる従来の問題点を解決するために本発明では図 1 5 のようなデータ列情報構造を提案する。

【 0 0 5 3 】 図 1 5 は本発明で提案する付加情報をデータ列情報(SOBI, Stream Object Information)領域で具現した例である。ここでは付加情報をCTフラグ(Contiguous flag)と称する。このCTフラグとTEフラグとはセル情報に貯蔵されることもある。分離されているデータ列が連続であることを知らせるCTフラグのない場合、分離されたデータ列を統合して分離前の状態に復元できない。

50 い。なぜなら、分離された状態で従来の探索情報のみで

は隣接したデータ列が統合可能なのか分からないからである。従って、隣接したデータ列が分離される前に連続、再生可能な状態であったか否かを知らせる情報が完全復元を行うために必要である。

【0054】従って、CTフラグは別の探索情報を構成したり、既在の探索情報の何処でも位置できるが、本発明の実施例ではデータ列情報に位置している。これは臨時削除というのがデータ列に対して行われるもので、CTフラグはデータ列に発生した変化を反映するものであるからである。従って、図15では臨時削除を示すTEフラグと臨時削除以前に連続的な1つのデータ列であったことを知らせるCTフラグをデータ列情報に貯蔵する。

【0055】CTフラグは、CTフラグと関連したデータ列が先行データ列と臨時削除以前に連続的な1つのデータ列であったことを意味する。これは先行データ列と統合可能なものを意味する。これは1つのデータ列を部分削除して分離する場合に主に発生する。もちろん、1つのデータ列情報に先行データ列と連続されたことを示すCTフラグと次のデータ列と連続されたことを示すCTフラグとを全て表現することもできる。このような構造は情報を一個所に集めて管理するという長所はあるが、各データ列に先行データ列と関連した情報を1つずつ置くことより規則性が劣ってむしろ管理しにくい。従って、ここでは1つのデータ列に1つのCTフラグが存する場合に対して説明する。

【0056】図16は分離されたデータ列に図15のデータ列情報を付加える例であって、本発明では図6と比較すれば、臨時削除で分離されたデータ列のためのSOBIにCTフラグを追加している点が相違である。セル2のSOB2が臨時削除された部分を示すのでSOBI2にはTEフラグがセットされている。また、SOBI2には先行データ列のSOB1と臨時削除以前に1つの連続されたデータ列であったからCTフラグもセットされている。部分削除で分離されたSOB3に関連したSOBI3にはSOB2と臨時削除以前に1つの連続されたデータ列であったからCTフラグがセットされている。SOB3は臨時削除されたデータ列でないため、TEフラグはセットされていない。

【0057】図17は本発明において提案するCTフラグを用いて図16の分離されたデータ列を復元する例であって、従来の復元方法(図14)を使用した場合には復元されなかった「\*」で表されたxから1.9まで、3.5からyまでの領域が完全復元されることがわかる。

【0058】本発明において提案する復元過程は次の通りである。

第1段階:完全復元過程で行われる統合過程の対象となるデータ列を決定する。ここで、説明の便宜上、臨時削除されて復元するセルを目標セル(target cell)と称し、この目標セルに対応するデータ列を目標データ列(target SOB)と称する。図16では目標セルはセル2で、目標データ列はSOB2となる。

【0059】完全復元過程が始まると、まず何れのデータ列を統合するかを決定すべきである。この過程はTEフラグとCTフラグとを用いて解決でき、復元過程の適用範囲は図18に示されている。完全復元は目標データ列から始まる。目標データ列のTEフラグとCTフラグとがセットされていると、この時先行データ列のTEフラグはリセット状態でなければならない。これは先行データ列が正常な状態であるべきことを意味する。

【0060】2つのデータ列のフラグが正常であることが確認されるとこれらを統合しうる。また、目標データ列の後続データ列のCTフラグがセット状態で、TEフラグがリセット状態ならばこれらは統合しうる。つまり、先行データ列、ターゲットデータ列、後続データ列を全て統合可能なのが類推できる。図16に示されたプログラムの完全復元時、CTフラグを活用すれば図18に示されたようにSOB1、SOB2、SOB3を全て統合できることがわかる。

【0061】第2段階:データ列情報とセル情報とに存する関連した探索情報の値を元の探索情報の値に復元する。図19は臨時削除前、削除後、そして完全復元後にデータ列と関連した探索情報などを示すテーブルである。

【0062】データ列復元時探索情報を更新する方法は次の通りである。

第1段階:SOB\_recの最初AP到達時間はSOB1の最初AP到達時間とする。

第2段階:SOB\_recの最後AP到達時間はSOB3の最後AP到達時間とする。

第3段階:SOB\_recの最初SOBUはSOB1の最初SOBUとする。SOB1、SOB2、SOB3を統合する場合に最初SOBUは統合前、最初SOB1の最初SOBUとなり、図20は第1段階乃至第3段階を説明するための図面である。

【0063】第4段階:SOB\_recのMAPL項目数はSOB1、SOB2、SOB3のMAPL項目数を全て足した値とする。本発明の実施例(図16)ではSOB1、SOB2、SOB3が全て1つのSOBUを有するのでSOB1、SOB2、SOB3を統合する場合、SOBU数はこれらのそれぞれのデータ列の有するSOBUを全て合算したものとなる。

第5段階:SOB\_recのIAPATを修正する。これに対しては図21、図22A、図22B、図23A、図23B、図24及び図25を結び付けて説明する。

【0064】図21は統合前、後のデータ列(SOB)と各SOBに対応する連結目録(MAPL)とIAPATを示す。IAPATは対応するSOBUの位置に応じて他の値を有する。これはIAPATに対する定義がSOBUの位置によって変わるからである。統合前にはSOBU(1)はSOB1の最後SOBUで、SOBU(2)がSOB2の最初SOBUである。また、SOBU(2)はSOB2の最後SOBUで、SOBU(3)はSOB3の最初SOBUである。従ってSOBU(1)のIAPATは1.0が増えた3.0となり、SOBU(2)のIAPATは1.0が減った1.0となり、SOBU(3)の

IAPATは1.0が減った1.0となる。

【0065】まず、SOB1の最後SOBUのSOBU(1)のIAPAT、そしてSOB3の最初IAPATを修正する。図21ではSOB2が1つのSOBUからなるためSOB2の最初と最後SOBUは同一である。統合後、SOB1とSOB2の境界で、SOB1の最後SOBUのSOBU(1)と関連したIAPATとSOB2の最初SOBUのSOBU(2)と関連したIAPATはデータ列の内部における相対的な位置が変更される。従って、これらの値を修正すべきである。

【0066】まず、SOB1の最後SOBUのSOBU(1)のIAPATを修正する方法の流れ図は図22Aに示された通りである。まず、SOB1の最後SOBUのSOBU(1)の最後AP到達時間の整数部分に1を足し、これをpreEnd\_high(=2)という変数に貯蔵し、SOB2の最初AP到達時間をsucStart\_APAT(=2.1)に貯蔵し、SOB2の最初AP到達時間から整数部分のみ取ったことをsucStart\_APAT\_high(=2.0)に貯蔵し、SOB1の最後SOBUのSOBU(1)のIAPATをpreEnd\_IAPAT(=2)という変数に貯蔵する(S201段階)。sucStart\_APAT\_high値とpreEnd\_high値との差(delta=0)を求め、preEnd\_IAPAT(=2)に差(delta)を足す(S202段階)。sucStart\_APATが整数なのかを判断し(S203段階)、もしsucStart\_APATが整数でなければpreEnd\_IAPAT(=2)に1を足して修正する(S204段階)。修正されたpreEnd\_IAPAT(=3)をSOB1の最後SOBUのIAPATに貯蔵したり、S203段階でsucStart\_APATが整数ならばS202段階で得られたpreEnd\_IAPAT(=2)をSOB1の最後SOBUのIAPATに貯蔵する(S205段階)。従って、SOBU(1)のIAPATは3となる。

【0067】次いで、統合後SOB2の最初SOBUのSOBU(2)のIAPATを修正する方法の流れ図は図23Aに示されたようである。まずSOB2の最初SOBUのSOBU(2)の最初AP到達時間をsucStart\_APAT(=2.1)という変数に貯蔵し、SOB2の最初SOBUのSOBU(2)のIAPATをsucStart\_IAPAT(=2)という変数に貯蔵する(S221段階)。sucStart\_APATが整数なのかを判断し(S222段階)、もしsucStart\_APATが整数でなければsucStart\_IAPAT(=2)を1だけ減少させ修正する(S223段階)。修正されたsucStart\_IAPAT(=1)をSOBU(2)のIAPATに貯蔵したり、S222段階でsucStart\_APATが整数ならばS221段階で設定されたsucStart\_IAPAT(=2)をそのままSOB2の最初SOBUのIAPATに貯蔵する(S224段階)。従って、SOBU(2)のIAPATは1となる。

【0068】図24はSOB1とSOB2にデータ列が統合される場合に、図22A及び図23Aで使用した変数が指示する値を図式化している。同様に、統合後SOB2とSOB3の境界でもSOB2の最後SOBUのSOBU(2)と関連したIAPATとSOB3の最初SOBUのSOBU(3)と関連したIAPATはデータ列の内部における相対的な位置が変更される。従って、これら値を修正すべきである。

【0069】まず、SOB2の最後SOBUのSOBU(2)のIAPAT

を修正する方法の流れ図は図22Bに示されたようである。まずSOB2の最後SOBUのSOBU(2)の最後AP到達時間の整数部分に1を足し、これをpreEnd\_high(=4)という変数に貯蔵し、SOB3の最初AP到達時間をsucStart\_APAT(=3.5)に貯蔵し、SOB3の最初AP到達時間から整数部分のみ取ったことをsucStart\_APAT\_high(=3.0)に貯蔵し、SOB2の最後SOBUのSOBU(2)のIAPATをpPreEnd\_IAPAT(=1)という変数に貯蔵する(S211段階)。sucStart\_APAT\_high値とpreEnd\_high値との差(delta=-1)を求め、preEnd\_IAPAT(=1)に差(delta)を足す(S212段階)。sucStart\_APATが整数なのかを判断し(S213段階)、もしsucStart\_APATが整数でなければpreEnd\_IAPATに1を足して修正する(S214段階)。修正されたpreEnd\_IAPAT(=1)をSOB2の最後SOBUのIAPATとしたり、S213段階でsucStart\_APATが整数ならばS212段階で設定したpreEnd\_IAPAT(=0)をSOB2の最後SOBUのIAPATに貯蔵する(S215段階)。従って、SOBU(1)のIAPATは1となる。

【0070】次いで、統合後SOB3の最初SOBUのSOBU(3)のIAPATを修正する方法の流れ図は図23Bに示されている。まず、SOB3の最初SOBUのSOBU(3)の最初AP到達時間をsucStart\_APAT(=3.5)という変数に貯蔵し、SOB3の最初SOBUのSOBU(3)のIAPATをsucStart\_IAPAT(=2)という変数に貯蔵する(S231段階)。sucStart\_APATが整数なのかを判断し(S232段階)、もしsucStart\_APATが整数でなければsucStart\_IAPAT(=2)を1だけ減少させ修正する(S233段階)。修正されたsucStart\_IAPAT(=1)をSOBU(3)の最初SOBUと関連したIAPATに貯蔵したり、S232段階でsucStart\_APATが整数ならばS231段階で設定されたsucStart\_IAPAT(=2)をSOBU(3)の最初SOBUと関連したIAPATに貯蔵する(S234段階)。従って、SOBU(3)のIAPATは1となる。

【0071】図25はSOB2とSOB3とにデータ列が統合される場合に図22B及び図23Bで使用した変数が指示する値を図式化している。

第6段階:SOB\_recの状態(status)を決定する。完全復元された状態なのでSOB\_recのTEフラグとCTフラグとをリセットする。

第7段階:SOB\_recに該当するセルで最初APの到達時間をSOB1に該当するセルの最初AP到達時間とする。

第8段階:SOB\_recに該当するセルにおいて最後APの到達時間をSOB3に該当するセルの最後AP到達時間とする。

【0072】図26はxからyに該当するデータ列を永久削除した例であって、この時永久削除可能なSOBUはSOBU(2)である。永久削除を行う場合には、探索情報だけでなく連結情報も変更すべきである。従って、xからyまでを永久削除する時、SOBU(1)は分離生成された新たなデータ列の最後SOBUなので該当IAPAT情報が変更され、SOBU(3)は分離生成された新たなデータ列の最初SOBUなので該当IAPAT情報が変更されるべきである。また、xから

yの間に存在する完全削除可能なSOBUのSOBU(2)の探索情報は削除され、SOBU(2)と関連した連結情報も削除されるべきである。参考にSOBUの一部のみ削除された場合は探索情報に残っている。このようにSOBU(2)と関連した探索情報と連結情報とが完全に削除されると、図26に示されたようにProgram\_orgの探索情報と連結情報はセル1とセル2の探索情報と連結情報として更新(update)される。

【0073】図27は本発明に係る永久削除によるデータ列分離時、先行SOB(図26のセル1に該当)の最後SOBUと関連したIAPATを修正する方法を示す流れ図である。永久削除によるデータ列分離後SOB1とSOB2の境界で、SOB1の最後SOBUのSOBU(ix=1)と関連したIAPATはSOBU(ix)のデータ列の内部における相対的な位置が変更されるのでこの値を修正すべきである。SOBU(ix)の最後AP到達時間、SOBU(ix+1)の最初AP到達時間、SOBU(iy-1)の最後AP到達時間、SOBU(iy)の最初AP到達時間は既知のことと仮定する。

【0074】まず、SOB2の最初SOBUのSOBU(ix+1=2)の最初AP到達時間を丸めて整数にし、これをpreEnd\_high(=3)という変数に貯蔵し、SOB1の最後SOBUのSOBU(ix=1)の最後AP到達時間の整数分のみ取ったものをpreEnd\_APAT\_high(=1)という変数に貯蔵し、SOB1の最後SOBUのSOBU(ix)のIAPATをpreEnd\_IAPAT(=3)という変数に貯蔵する(S301段階)。preEnd\_high値とpreEnd\_APAT\_high値との差(delta=2)を求め、preEnd\_IAPAT(=3)から差(delta=2)を引き、その減算結果(=1)を1だけ増加させてpreEnd\_IAPATを修正する(S302段階)。修正されたpreEnd\_IAPAT(=2)をSOBU(ix)のIAPATに貯蔵する(S303段階)。従って、SOBU(ix=1)のIAPATは2となる。

【0075】図28は本発明に係る永久削除によるデータ列お分離時、後続SOB(図26のセル2に該当)の最初SOBUと関連したIAPATを修正する方法を示す流れ図である。SOB2の最初SOBUのSOBU(iy=3)と関連したIAPATはデータ列の内部における相対的な位置が変更されるのでこの値を修正すべきである。

【0076】まず、SOB2の最初SOBUのSOBU(iy=3)の最初AP到達時間をsucStart\_APAT(=3.5)という変数に貯蔵し、SOB2の最初SOBUのSOBU(iy)のIAPATをsucStart\_IAPAT(=1)という変数に貯蔵する(S311段階)。sucStart\_APATが整数なのかを判断し(S312段階)、もしsucStart\_APATが整数でなければsucStart\_IAPAT(=1)を1だけ増加させ修正する(S313段階)。修正されたsucStart\_IAPAT(=2)をSOBU(iy)のIAPATに貯蔵したり、S312段階でsucStart\_APATが整数ならばS311段階で設定したsucStart\_IAPAT(=1)をそのままSOBU(iy)のIAPATに貯蔵する(S314段階)。従って、SOBU(iy=3)のIAPATは2となる。本発明はデジタル貯蔵媒体を用いる記録/再生装置に適用でき、特にストリームレコーダに効

率よく適用しうる。

【0077】

【発明の効果】本発明は臨時削除によるデータ列の分離時、従来に提示されなかった探索情報を生成する方法を提示し、本発明の情報構造と復元方法を用いると、臨時削除で分離されたデータ列を臨時削除される前の状態に完全に復元させうる。また、本発明には探そうとするデータが属しているデータ列ユニットを探し出す過程でSOBUの累積時間(IAPAT)を累算する時、初期値を“0”にせず、データ列の最初データ到達時間の整数部に設定することによって高速探索が可能である。また、臨時削除された部分を永久削除する場合にも探索情報のみでも削除範囲が分かるために迅速な削除が要求される場合に効率よく使用しうる。

【図面の簡単な説明】

【図1】 従来の所望の部分のデータを読み出すのに用いられる探索情報とデータとの間の関係を示す一例である。

【図2】 図1に示された探索情報におけるAPAT及び／またはIAPATの表示に用いられる領域を示す図面である。

【図3】 従来のIAPATを求める過程を説明するための図面である。

【図4】 図4Aと図4Bは境界領域におけるIAPATを求める例である。

【図5】 本発明の理解のため、データの一部を臨時削除するために削除範囲を決定する過程を示す例である。

【図6】 臨時削除で分離されたデータ列の例である。

【図7】 分離された各セルのそれぞれの探索情報を示すテーブルである。

【図8】 分離前後の各セルのそれぞれの探索情報の変化を示す図面である。

【図9】 分離前後の各セルのそれぞれのIAPAT変化を示す図面である。

【図10】 図10Aと図10Bは本発明に係るデータ列の分離時、先行SOBの最後SOBUと関連したIAPAT修正方法を示す流れ図である。

【図11】 図11Aと図11Bは本発明に係るデータ列の分離時、後続SOBの最初SOBUと関連したIAPAT修正方法を示す流れ図である。

【図12】 SOB1とSOB2とにデータ列が分離される場合、図10Aと図11Aで使用した変数が指示する値を説明するための図面である。

【図13】 SOB2とSOB3とにデータ列が分離される場合、図10Bと図11Bで使用した変数が指示する値を説明するための図面である。

【図14】 単純復元方法で臨時削除により分離されたデータ列を復元する例である。

【図15】 本発明に係るSOBI構造の一例である。

【図16】 図15に示されたSOBI情報を用いてデータ



列の分離を示す実施例である。

【図 17】 図 15 に示された SOBI 情報を用いて分離されたデータ列を復元する例である。

【図 18】 本発明に係る復元過程の適用範囲を示す一例である。

【図 19】 臨時削除前、後、そして完全復元後にデータ列と関連した探索情報を示すテーブルである。

【図 20】 統合前後の各 SOB の AP 到達時間を説明するための図面である。

【図 21】 統合前後のデータ列とこれに対応する連結目録 (MPAL) とデータ列ユニットの持続時間 (IAPAT) との関係を示す一例である。

【図 22】 図 22A 及び図 22b はデータ列の統合時、先行データ列の最後データ列ユニットと関連した IAPAT 修正方法を示す流れ図である。

【図 23】 図 23A 及び図 23B はデータ列の統合時、

後続データ列の最初データ列ユニットと関連した IAPAT 修正方法を示す流れ図である。

【図 24】 SOB1 と SOB2 とにデータ列が統合される場合、図 22A と図 23A で使用した変数が指示する値を示す図面である。

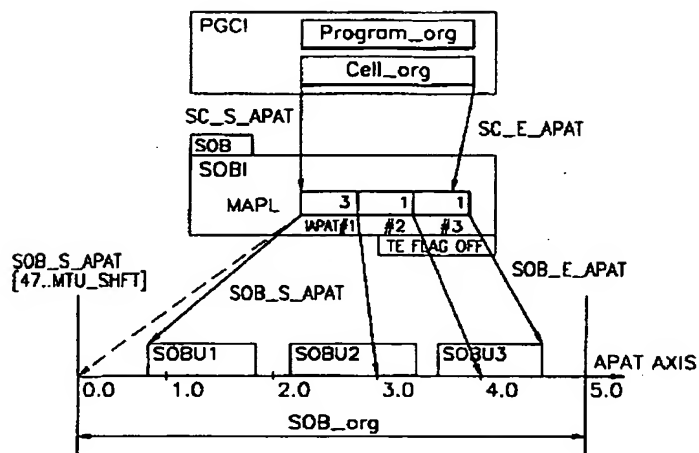
【図 25】 SOB2 と SOB3 とにデータ列が統合される場合、図 22B と図 23B で使用した変数が指示する値を示す図面である。

【図 26】 プログラムの一部を永久削除する一例である。

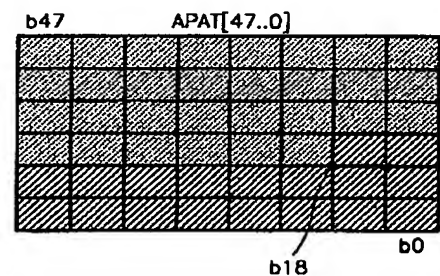
【図 27】 本発明に係る永久削除によりデータ列の分離時、先行データ列の最後データ列ユニットと関連した IAPAT を修正する方法を示す流れ図である。

【図 28】 本発明に係る永久削除によるデータ列の分離時、後続データ列の最初データ列ユニットと関連した IAPAT を修正する方法を示す流れ図である。

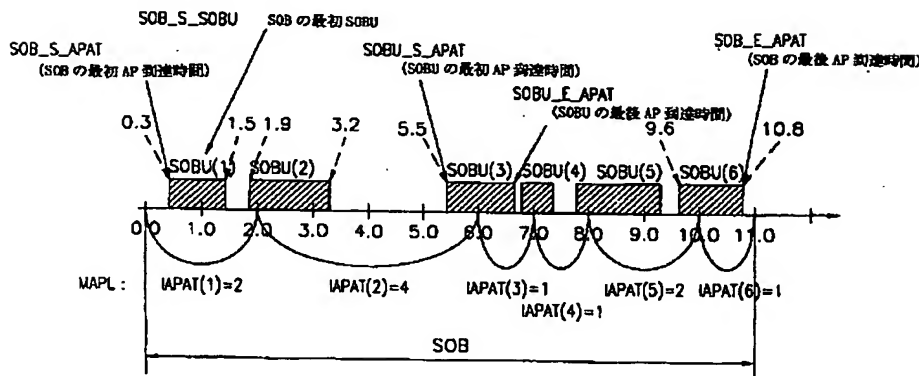
【図 1】



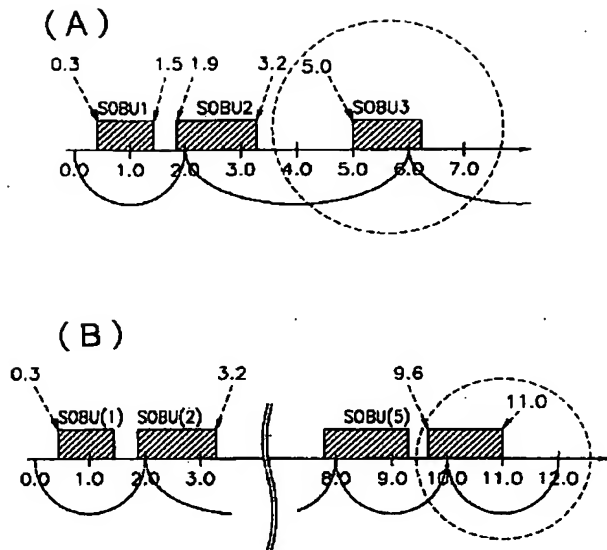
【図 2】



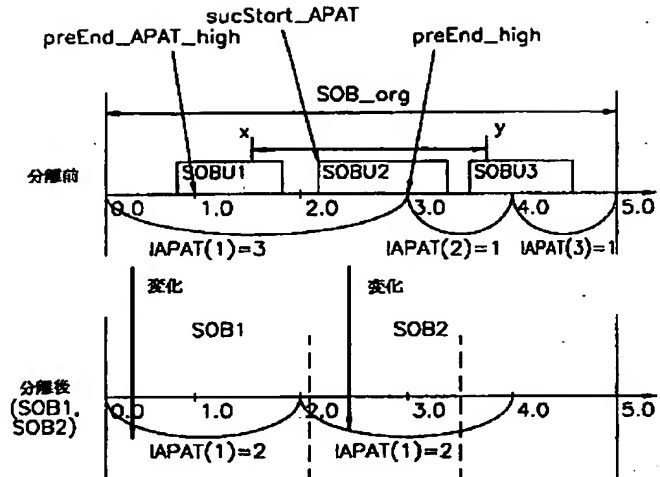
【図 3】



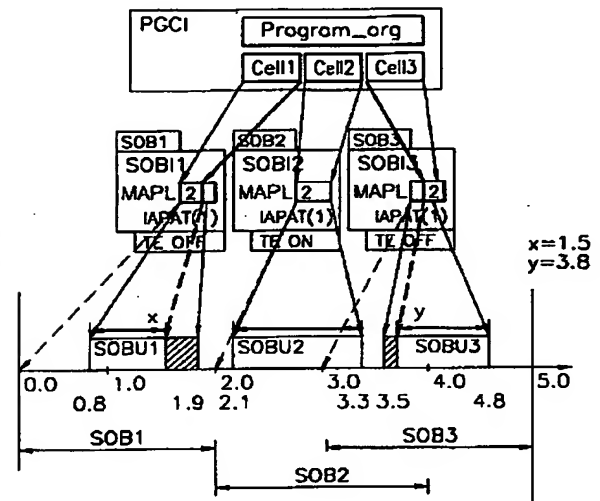
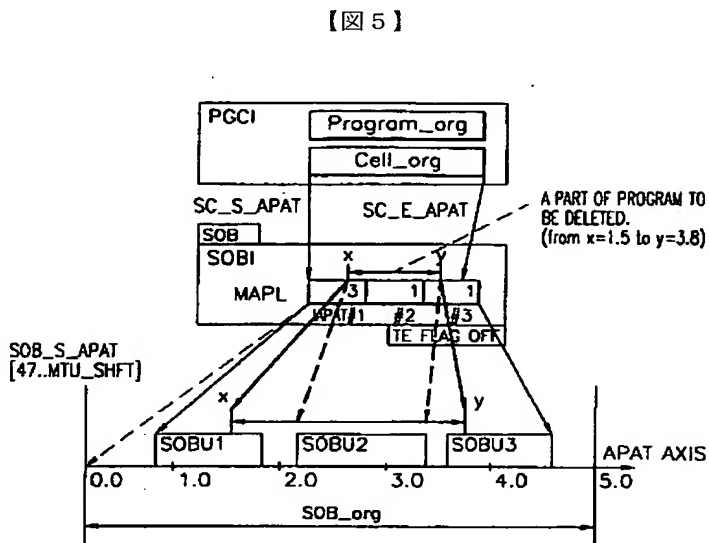
【図 4】



【図 12】



【図 14】

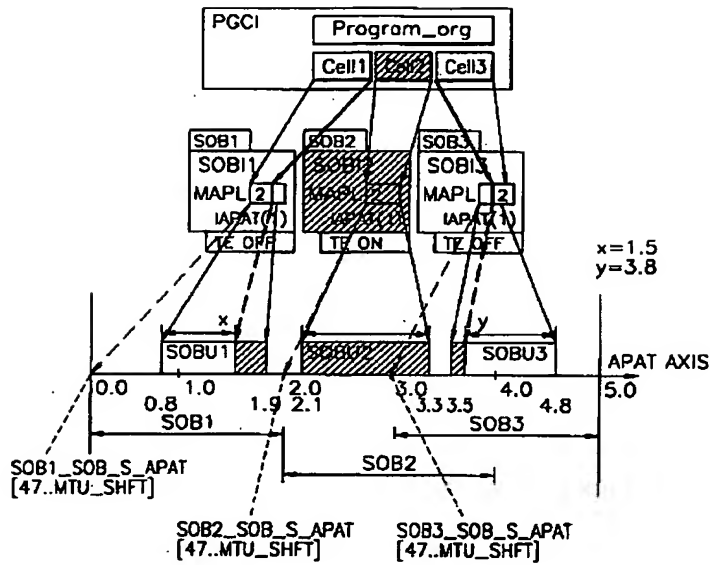


【図 7】

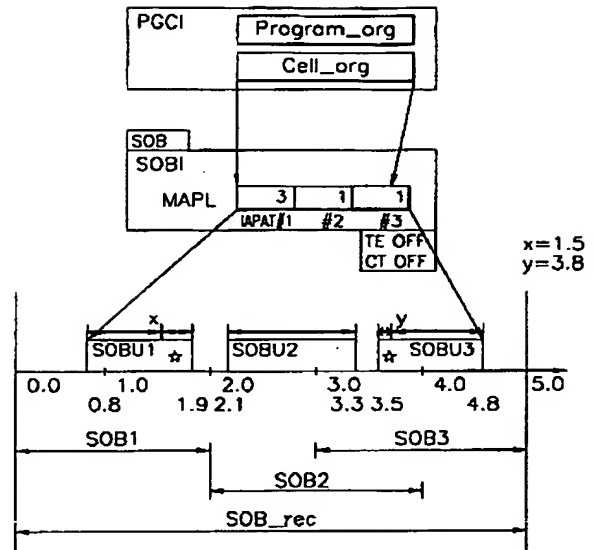
	SOB_S _APAT	SOB_E _APAT	SOB_S _SOBU	MAPL ENT_Ns	MAPL [IAPATs]	SOB_TY	SC_S _APAT	SC_E _APAT
部分削除前、SOB	0.8	4.5	SOBU1	3	[3,1,1]		0.8	4.5
部分削除前、SOB1	0.8	1.9	SOBU1	1	[2]		0.8	1.5
SOB2	2.1	3.3	SOBU2	1	[2]	TE,CT	2.1	3.3
SOB3	3.5	4.8	SOBU3	1	[2]	CT	3.8	4.5



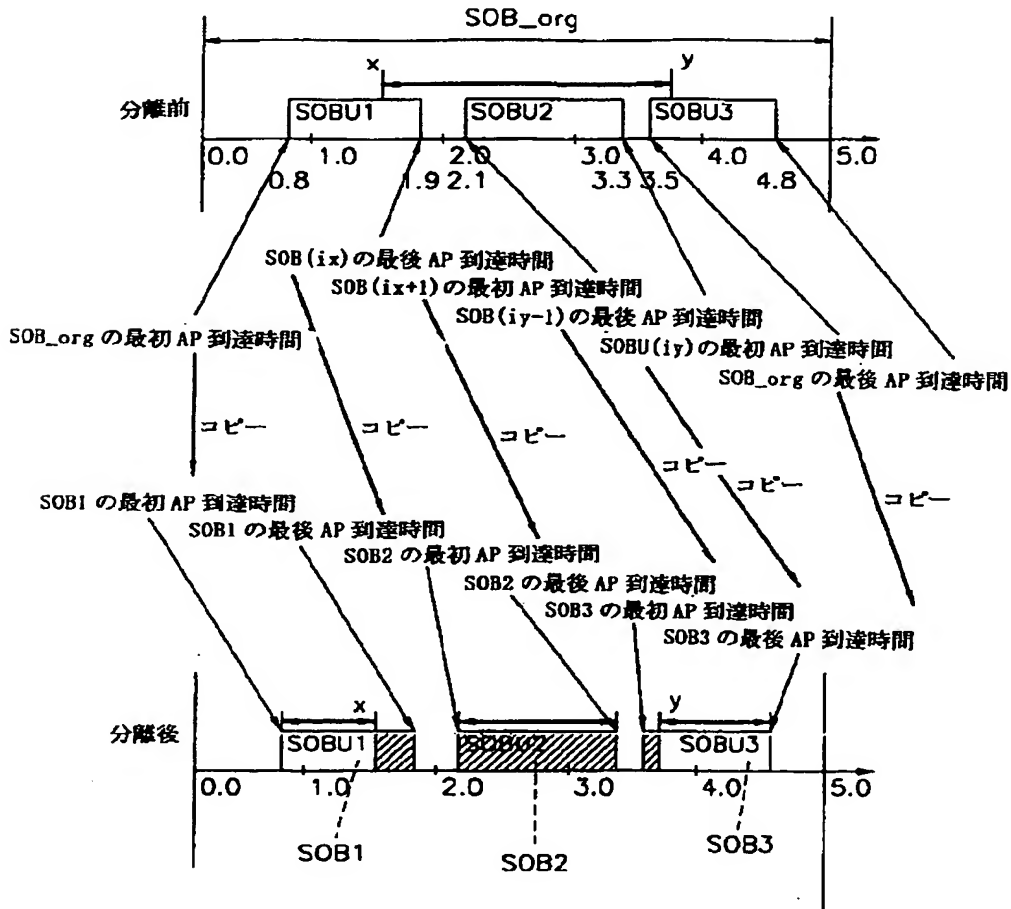
【図 6】



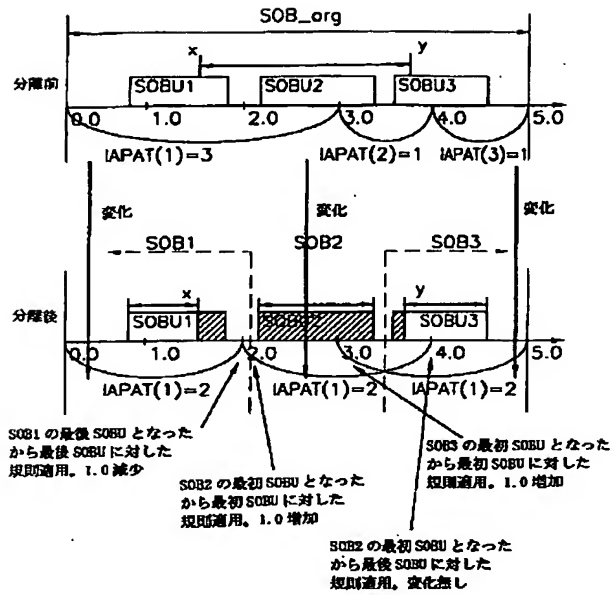
【図 17】



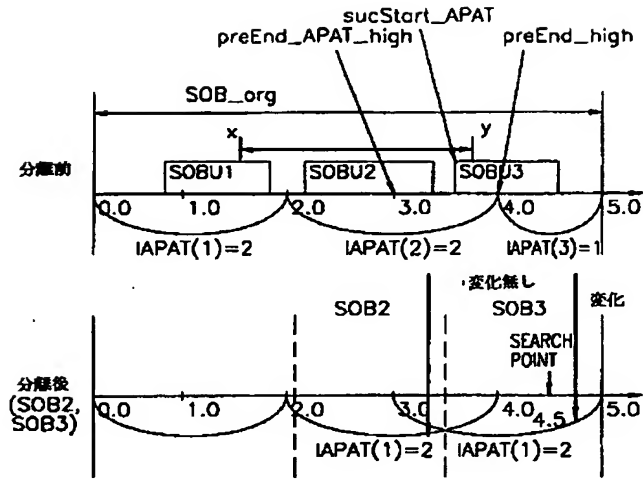
【図 8】



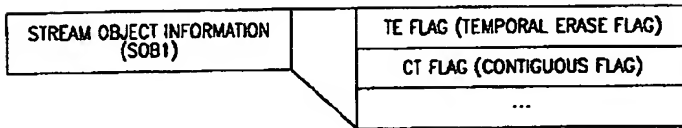
【図 9】



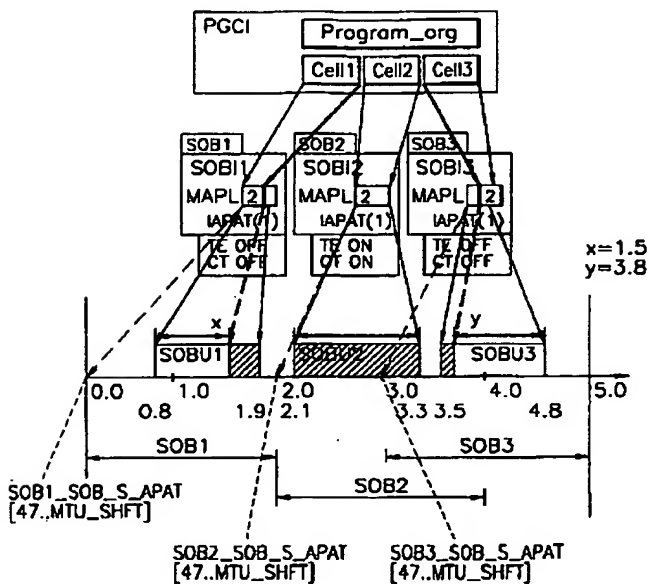
【図 13】



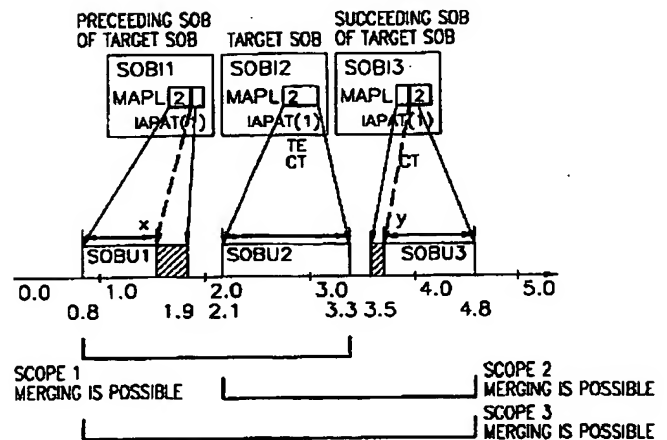
【図 15】



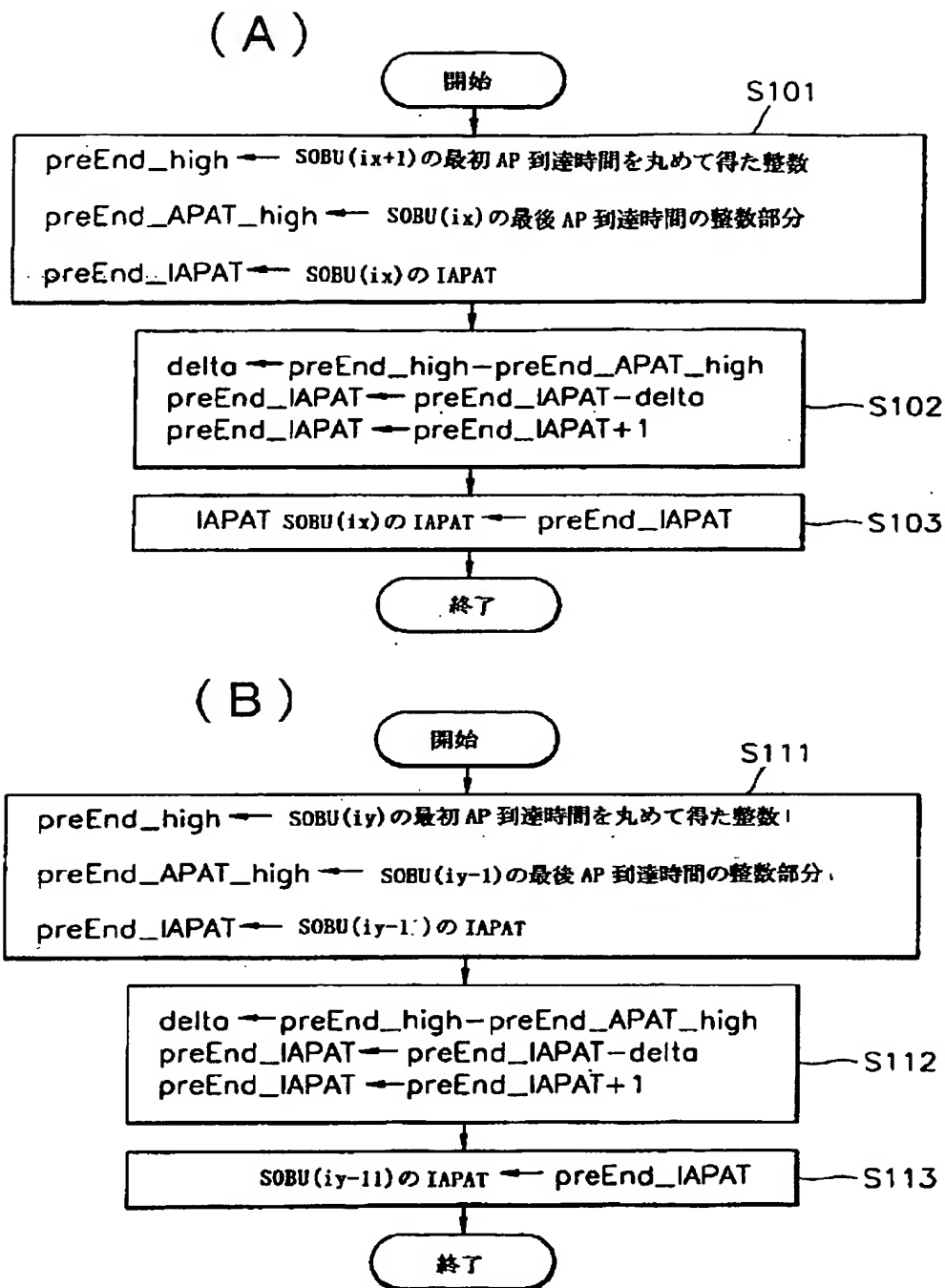
【図 16】



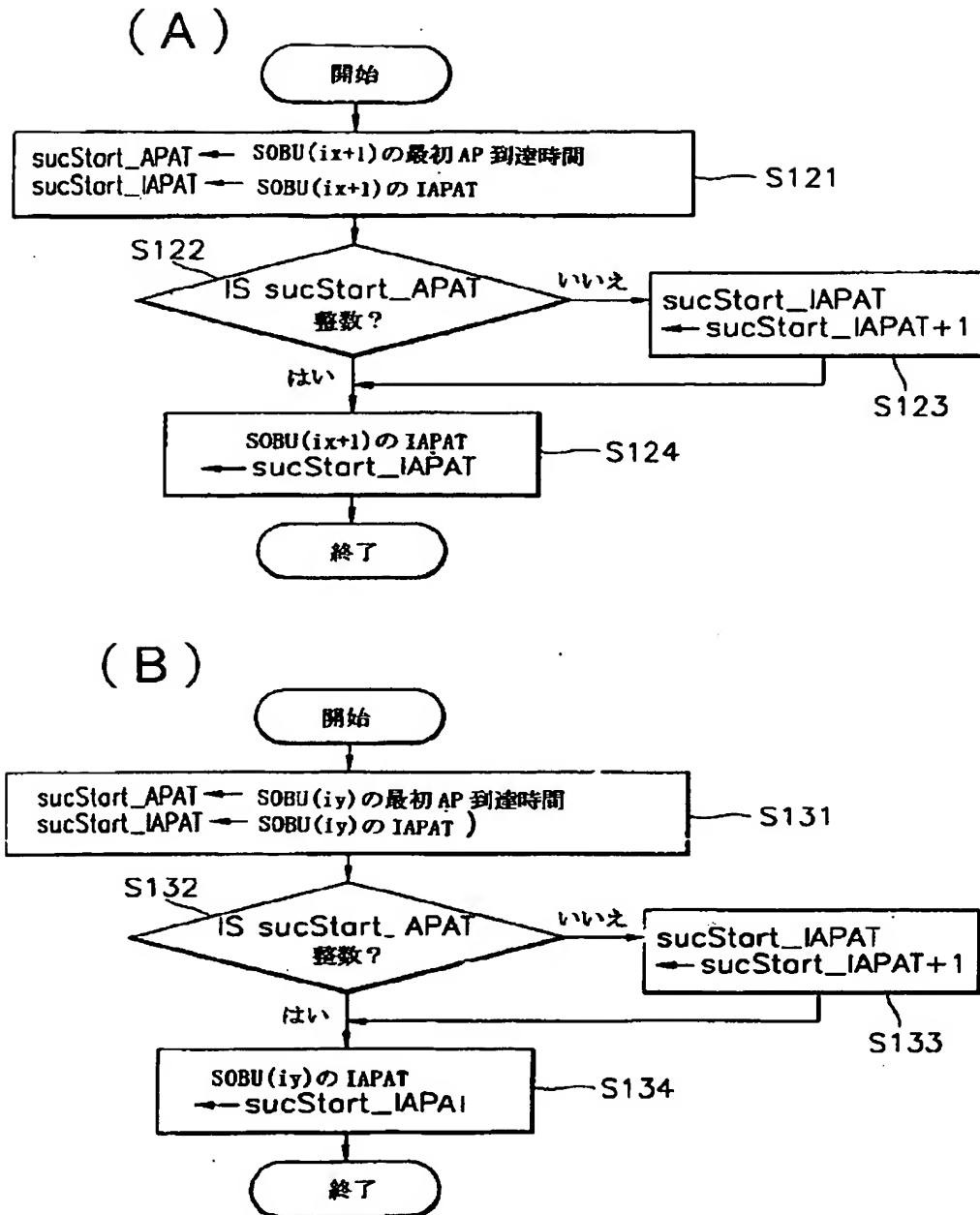
【図 18】



【図 10】



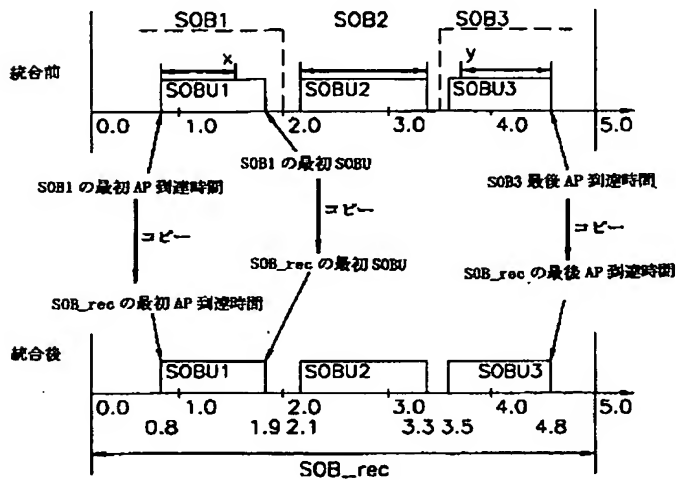
【図 1.1】



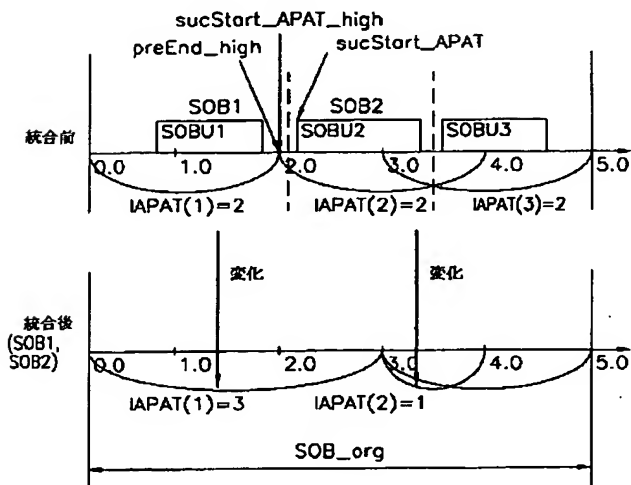
【図 19】

		SOB_S _APAT	SOB_E _APAT	SOB_S _SOBU	MAPL ENT_Ns	MAPL [IAPAT's]	SOB_TY	SC_S _APAT	SC_E _APAT
臨時削除前	SOB_ org	0.8	4.5	SOBU1	3	[3,1,1]		0.8	4.5
臨時削除後	SOB1	0.8	1.9	SOBU1	1	[2]		0.8	<u>1.5</u>
	SOB2	2.1	3.3	SOBU2	1	[2]	TE,CT	2.1	3.3
	SOB3	3.5	4.5	SOBU3	1	[2]	CT	<u>3.8</u>	4.5
完全復元後	SOB_ rec	0.8	4.5	SOBU1	3	[3,1,1]		0.8	4.5

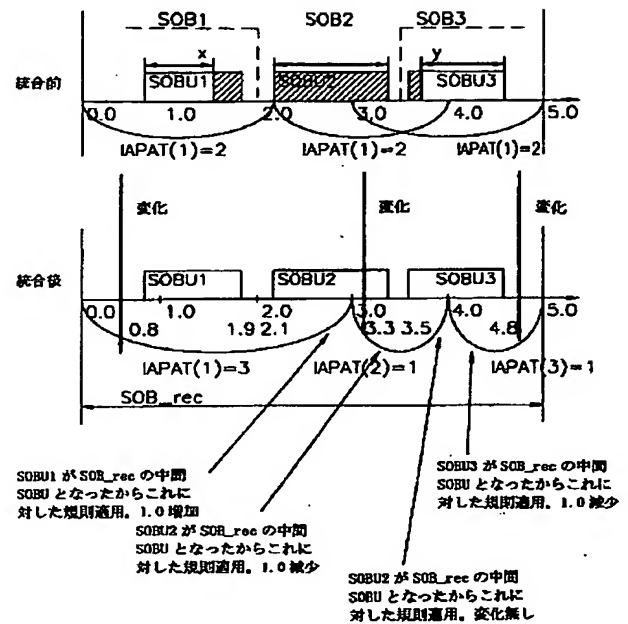
【図 20】



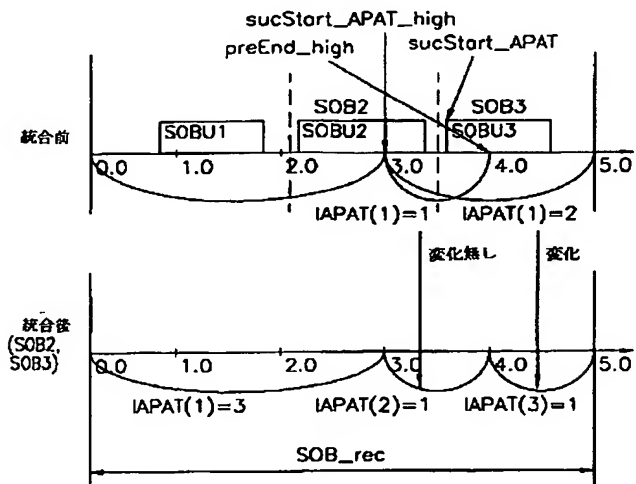
【図 24】



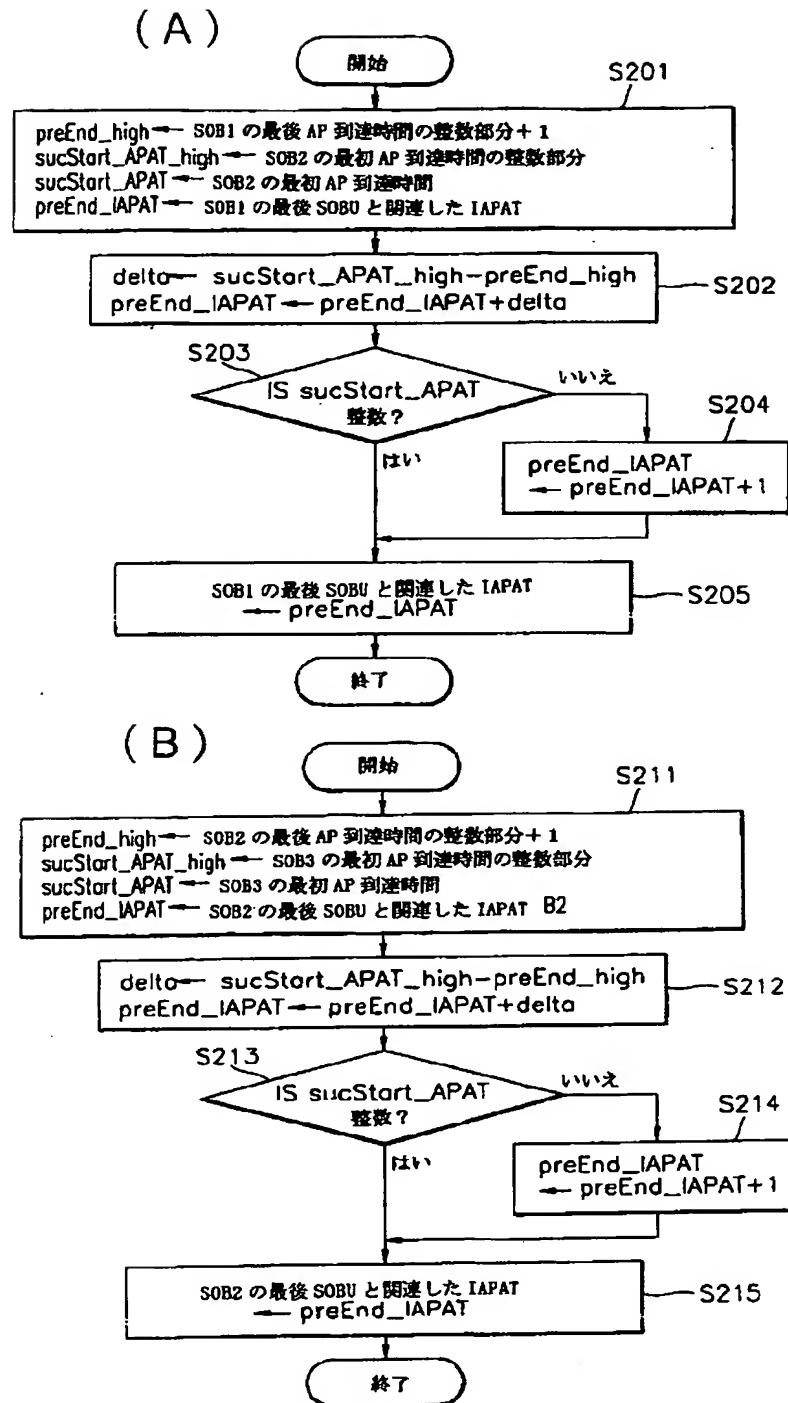
【図 21】



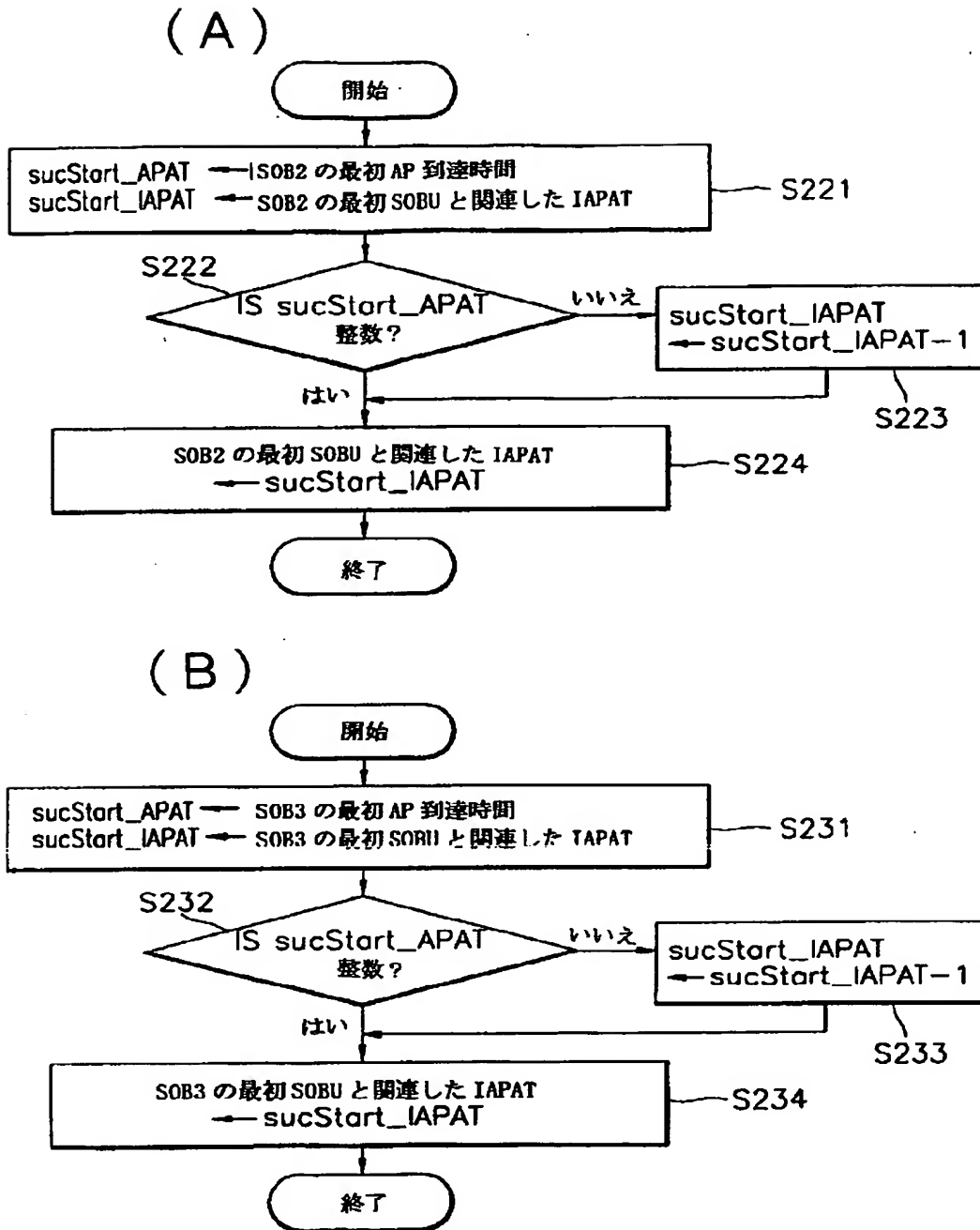
【図 25】



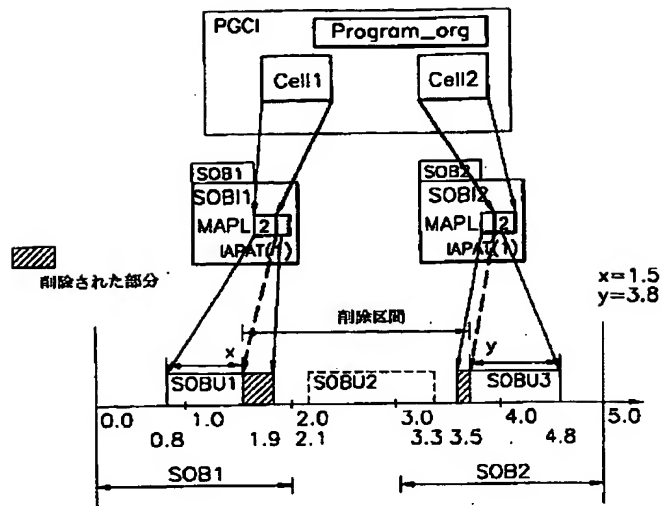
【図 2.2】



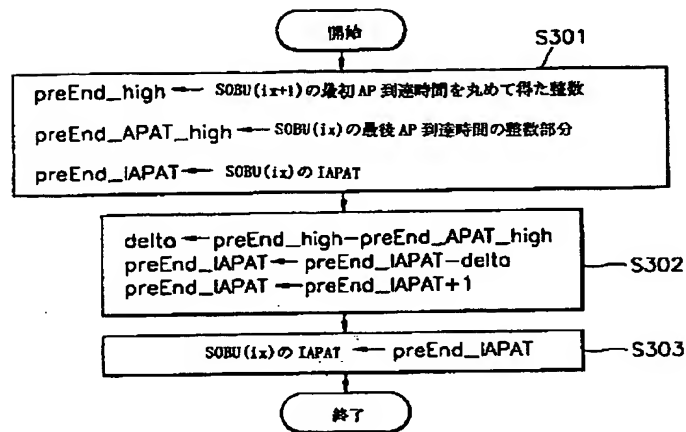
【図2.3】



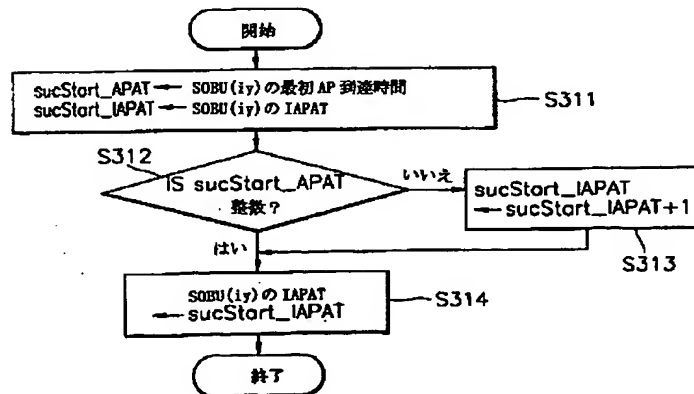
【図 2.6】



【図 2 7】



【図 2 8】





フロントページの続き

(72)発明者 吳 永南

大韓民国京畿道城南市盆唐区盆唐洞39番地  
セッビョルマウル403棟302号